Africa Centre of Excellence on Technology Enhanced Learning National Open University of Nigeria

A Secure Data-Centric Model for Digital Learning in Higher Institutions

by

Ismaila Idris Sinan
ACE21150003

Thesis Submitted to African Centre of Excellence on Technology Enhanced Learning National Open University of Nigeria for the Award of Doctorate in Cyber Security

Supervisor:
Dr Vivian Nwaocha

Co-Supervisors:
Prof. Jules Dégila
&
Prof. Adebukola Onashoga

Date of Submission:
22/12/2023

METADATA

Title: A Secure Data-Centric Model for Digital Learning in Higher Institutions

Student Name: Ismaila Idris Sinan

Supervisor: Dr Vivian Nwaocha

Co-Supervisors: Prof. Jules Dégila & Prof Adebukola Onashoga

Department: Africa Centre of Excellence on Technology Enhanced Learning

Qualification: Doctorate in Cyber Security

Institution: National Open University of Nigeria

Keywords: Data-centric model, security model, comparative analysis, proof of concept

Document Date: 22/12/2023

Sponsor: Digital Science and Technology Network

DECLARATION

I, Ismaila Idris Sinan ACE21150003, affirm that I conducted this thesis independently, and it has not been submitted for the fulfilment of any academic prerequisites or awards by any other means.

Ismaila Idris Sinan _____ 21/12/2023

Student Name Signature Date

Signature of the Supervisor

I, Vivian Nwaocha, herewith declare that I accept this thesis for my supervision.

Signature  Data: 21/12/2023

Signature of Co-Supervisors:

I, Jules Dégila, herewith declare that I accept this thesis for my supervision

Signature_____. Data: 17/01/2024

I, Adebukola Onashoga, herewith declare that I accept this thesis for my supervision.

Signature_____. Data: 18/01/2024

DEDICATION

This work is dedicated to the cherished memory of my Late Dad, Ash-Sheikh Ismaila Idris Ibn Zakariya, whose wisdom, and support have been an enduring source of inspiration. His influence continues to shape my journey. Additionally, I dedicate this work to my beloved Mother, whose love and encouragement have been my pillars of strength. May the soul of my dear Dad rest in peace, and may this work reflect the values he instilled in me.

ACKNOWLEDGEMENTS

I express my profound gratitude to my esteemed supervisors, Dr Vivian Nwaocha, Prof. Jules Dégila, and Prof. Adebukola Onashoga, for their exceptional guidance, unwavering support, and valuable insights throughout the entire research journey. The depth of their expertise and commitment significantly contributed to the success of this work.

A special acknowledgement is extended to Prof. Valerie Viet Triem Tong, my internship supervisor, and Prof. Grace Jokthan, Centre Director, for their mentorship and encouragement, which played a pivotal role in shaping the research direction.

I am grateful to my family members, including my brothers and sisters, whose encouragement and understanding provided a strong foundation. I extend my appreciation to my colleagues at work, friends, especially my wife Hajara Abubakar Maihali, and children Suwaiba Sinan, Rufaida Sinan, and Nusaiba Sinan, for their unwavering support.

I would like to acknowledge the financial support from Digital Science and Technology Network (DSTN) Confidentiality, Integrity, Disponibilité, Repartition (CIDRE) team, Africa Centre of Excellence on Technology Enhanced Learning (ACETEL), Institut de Recherche pour le Développement (IRD), and the invaluable contributions of Prof. Goussou Camara, Sopho, Helen, and my proposed wife Amina Aliyu Abdullahi. Their support and encouragement have been instrumental in the completion of this research.

Furthermore, I appreciate the collaborative efforts of the research community, including the DSTN, CIDRE team, ACETEL, IRD, for their financial support, which greatly facilitated the successful execution of this research.

In essence, the collective support, mentorship, and encouragement from these individuals and institutions have been crucial to this research's completion, and I am sincerely grateful.

LIST OF PUBLICATIONS FROM THE THESIS

1. I. I. Sinan, J. Degila, V. Nwaocha and S. A. Onashoga, "Data Architectures' Evolution and Protection," *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 2022, pp. 1-6, doi: 10.1109/ICECET55527.2022.9872597. https://ieeexplore.ieee.org/document/9872597
2. I. I. Sinan, V. Nwoacha, J. Degila and S. A. Onashoga, "A Comparison of Data-Driven and Data-Centric Architectures using E-Learning Solutions," 2022 International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS), Bandung, Indonesia, 2022, pp. 1-6, doi: 10.1109/ICADEIS56544.2022.10037358.https://ieeexplore.ieee.org/document/10037358
3. I. I. Sinan, V. Nwaocha, J. Degila and S. A. Onashoga, E-Learning Digitalization' Evolution and Transformation. Proceedings of the 33rd ECOWAS iSTEAMS Emerging Technologies, Scientific, Business, Social Innovations & Cyber Space Ecosystem Multidisciplinary Conference. University of Ghana/Academic City University College, Ghana. 29th Sept – 1st Oct, 2022. Pp 119-129. www.isteams.net/ecowasetech2022. dx.doi.org/10.22624/AIMS-/ECOWASETECH2022P23.https://www.isteams.net/_files/ugd/185b0a_41b1497099c2407aad194f6782a92eda.pdf
4. I. I. Sinan, V. Nwaocha, J. Degila and S. A. Onashoga, Data-architectures and the prevalent cyberattacks Encountered by West African Institutions in the COVID—19 Era, journal of Infrastructure, Policy and Development. **Accepted for publication**.
5. I. I. Sinan, V. Nwaocha, Valerie Viet Triem Tong, J. Degila and S. A. Onashoga, Enhancing Security and Privacy in Educational Environments: A Secure Grade Distribution Scheme with Moodle Integration, Journal of Infrastructure, Policy, and Development. **Accepted for publication.**
6. I.I. Sinan, Valérie Viet Triem Tong, V. Nwoacha, and J. Degila, "A Comprehensive Scheme for Secure Grade Distribution in Educational Settings: Integration of Cryptographic Techniques and User-Centric Security in Moodle," Under Review" European Interdisciplinary Cybersecurity Conference

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Digital learning universities face unique challenges in managing, securing, and extracting information from the vast amount of data generated in educational settings. This research focused on identifying the most suitable data architecture for these Universities in West Africa. The methodology involves a comparative analysis of 109 e-learning solution use cases, classified based on their data architectures. A total of 983 user reviews from the e-learning industry further inform the analysis, the study proceeds to develop a data-centric model specifically designed to meet the distinct needs of digital learning universities. The proposed data-centric model integrates essential components, such as data sources, a data hub, efficient data streaming through Apache Kafka, and data governance and security using Apache Ranger and Apache Atlas. The security framework embedded within the model employs Role-Based Access Control (RBAC), encryption through AES, and countermeasures to address identified threats. The research's innovation extends to a Secure Grade Distribution Scheme, practically implemented within the Moodle learning management system. This scheme leverages advanced features, including Diffie-Hellman key exchange, Hardware security module (HSM) and Message Integrity Code (MIC) verification, showcasing its adaptability and effectiveness in enhancing security within educational environments. The integrated proof of concept provides a practical demonstration of both the Secure Grade Distribution Scheme and the proposed Data-Centric Model within a controlled lab environment. This comprehensive approach ensures the validation of the research findings and their potential impact on the secure and efficient management of data in digital learning universities.

Keywords: Data-centric model, security model, comparative analysis, proof of concept

CHAPTER ONE: INTRODUCTION

1.1 Background of the Study

The integration of digital learning in higher education has become an imperative response to the changing educational landscape globally. The advent of the COVID-19 pandemic has acted as a catalyst, compelling educational institutions to swiftly adopt and adapt to digital methodologies to ensure the continuity of learning (Aulakh et al., 2023). This shift has, in turn, heightened the significance of robust data architectures to support the seamless functioning of digital learning platforms.

The challenges posed by the pandemic have been particularly pronounced in West African Universities, where the intersection of limited resources and a growing demand for digital education has amplified the complexities of the data management (Djeki et al., 2023). The vulnerabilities of existing data architectures have been exposed as universities grapple with the sudden surge in cyber-attacks targeting academic databases and communication channels (Sinan, Nwoacha, et al., 2022). These challenges necessitate a thorough investigation into the prevailing data architectures and their resilience to cyber threats during the pandemic.

Understanding the unique context of West African Universities is crucial in this study. These institutions often face resource constraints, both in terms of technological infrastructure and financial capabilities (Aborode et al., 2020; Bervell & Umar, 2017). This context adds layers of complexity to the task of ensuring data security in a digital learning environment. The dynamic nature of cyber threats and the evolving landscape of digital technologies further compound the challenges faced by these universities (Sun et al., 2023).

Scholarly discussions on the impact of digital learning in higher education have acknowledged the need for a nuanced understanding of the contextual factors influencing its implementation (Setiawan et al., 2023). The intersection of digital learning, data architecture, and cybersecurity in the specific context of West African Universities remains underexplored. This research aims to fill this gap by providing

insights that can inform both local and global strategies for enhancing data security in the rapidly evolving domain of higher education.

1.2 Statement of the Problem

The intersection of digital learning, data architectures, and cybersecurity in West African Universities presents a multifaceted challenge that demands comprehensive investigation. As higher education institutions rapidly transition to digital platforms, the vulnerabilities within existing data architectures become increasingly apparent, exacerbated by the unforeseen disruptions brought about by the COVID-19 pandemic. The surge in cyber-attacks targeting academic databases and communication channels has exposed critical shortcomings in the resilience of these architectures (Sinan, Degila, et al., 2022a; Sinan, Nwoacha, et al., 2022). This raises pressing concerns about the integrity, confidentiality, and availability of academic data essential for the smooth functioning of digital learning environments.

The challenges faced by West African Universities in this context are compounded by resource constraints, both in terms of technological infrastructure and financial capabilities (Aborode et al., 2020; Bervell & Umar, 2017). These constraints not only limit the ability of institutions to invest in sophisticated cybersecurity measures but also underscore the need for context-specific solutions that consider the unique socio-economic and technological landscape of the region. The absence of a dedicated exploration into the existing data architectures, cybersecurity challenges faced during the pandemic, and the countermeasures implemented by these universities leave a critical gap in understanding the holistic picture of data security in the digital learning domain.

Furthermore, the dynamic nature of cyber threats and the evolving landscape of digital technologies add an additional layer of complexity to the problem (Sun et al., 2023). There is a palpable urgency to address these challenges comprehensively, ensuring that any proposed solutions are not only effective in mitigating current threats but also adaptable to the evolving nature of cybersecurity risks in the higher education sector. Thus, the overarching problem to be addressed is the inadequacy of current data architectures in West African Universities to withstand cyber threats, particularly in the

context of the rapid shift towards digital learning platforms during and beyond the COVID-19 pandemic.

Considering these, this research seeks to investigate the following key aspects: the prevailing data architectures in West African Universities, the types of cyber-attacks faced during the pandemic, the countermeasures implemented by these universities, and ultimately, the design of a robust security model and a Secure Grade Distribution Scheme tailored to the specific needs of digital learning environments in the region.

1.3 Aim of the study

This study aims to provide insights into the prevailing data architectures, the types of cyber-attacks faced, and the countermeasures implemented, ultimately contributing to the development of a secure and resilient framework for digital learning.

1.4 Specific objectives

1. Comprehensive Survey of Existing Data Architectures, Cybersecurity Challenges, and Countermeasures in West African Universities during the COVID-19 Pandemic
    a. Conduct an extensive survey to identify prevailing data architectures within West African Universities.
    b. Analyse the types of cyber-attacks faced by these architectures during the COVID-19 pandemic.
    c. Identify and assess the countermeasures implemented by these universities to mitigate cybersecurity challenges.
2. Comparative Analysis for Optimal Digital Learning University Architecture Using E-learning Use Cases
    a. Conduct a rigorous comparative analysis of identified data architectures, focusing on e-learning use cases to determine the most suitable one for meeting the evolving demands of a digital learning university.
3. Design and Proposal of a Security Model for Digital Learning Universities
    a. Develop a robust security model tailored to address the specific threats identified in digital learning environments within universities.
    b. Propose enhancements to existing security measures to fortify data protection.

4. Development of a Secure Grade Distribution Scheme
    a. Create a Secure Grade Distribution Scheme specifically designed for digital learning universities, ensuring the integrity and confidentiality of students' grades.
5. Proof of Concept for Data Architecture and Grade Distribution Scheme
    a. Provide a practical demonstration of the proposed data architecture and grade distribution scheme to validate their effectiveness in a real-world digital learning setting.

1.5 Scope of the Study

This research will focus specifically on West African Universities, recognizing the unique contextual factors that influence digital learning, data architectures, and cybersecurity in this region. The scope encompasses a comprehensive exploration of existing data architectures, cybersecurity challenges faced during the pandemic, and the development of targeted solutions to enhance the overall security posture of digital learning environments.

1.6 Significance of the Study

The significance of this study lies in its potential to address critical gaps in understanding and fortifying data security within the context of West African Universities. As these institutions grapple with the challenges of digital learning adoption exacerbated by the COVID-19 pandemic, insights gained from this research can inform strategic decisions at various levels. By unravelling the intricacies of existing data architectures, cybersecurity challenges, and countermeasures implemented by universities, this study contributes valuable knowledge that can be leveraged by educational policymakers, administrators, and technologists. Furthermore, the findings hold broader implications for higher education globally, offering a nuanced understanding of the intersection between digital learning, data architecture, and cybersecurity.

1.7 Definition of Terms

1. Data-Centric Architecture: Refers to an approach in system design where data is the primary focus, and applications are built around the data. It emphasizes efficient data management, accessibility, and usability.
2. E-Learning: The use of electronic technologies to facilitate learning and education. It involves the use of computers, digital resources, and the internet to deliver educational content and support interactive learning experiences.
3. Hybrid Storage: A storage solution that combines different types of storage technologies, such as Network-Attached Storage (NAS), Storage Area Network (SAN), Cloud Storage, Local Storage, and Tape Storage, to optimize performance and cost efficiency.
4. Data Governance: The overall management of the availability, usability, integrity, and security of data used in an organization. It involves defining data-related policies, standards, and processes to ensure effective data management.
5. Security Model: A structured framework that defines and enforces security policies, measures, and controls to protect data, systems, and resources from unauthorized access, attacks, and potential threats.
6. Data as a Service (DaaS): A service-oriented approach that provides on-demand access to data, allowing users to access and utilize data without the need for extensive local storage and management.
7. TensorFlow: An open-source machine learning library developed by Google that facilitates the development and deployment of machine learning models. It is widely used for tasks such as data analysis, classification, and predictive modelling.
8. KConnect: A connector software utilized in data-centric architectures to establish seamless communication and integration between different data sources and systems.
9. Apache Ranger: An open-source tool that provides centralized security administration for Hadoop-based data systems. It enables fine-grained access control and security policies.

10. Apache Atlas: An open-source tool for metadata management and data governance in Hadoop-based ecosystems. It allows organizations to create, share, and manage metadata and data lineage information.
11. Kibana & Elastic Search: Tools used for creating user-centric applications. Kibana provides visualization capabilities, while Elastic Search offers efficient data retrieval, enhancing the user experience.
12. Kafka: A distributed streaming platform used for building real-time data pipelines and streaming applications. It ensures reliable and scalable data streaming between different components of a system.
13. Proof of Concept: A demonstration or experiment that validates the feasibility and functionality of a proposed model or system, typically involving simulated scenarios and real-world testing.
14. Digital Learning Universities: Institutions of higher education that leverage digital technologies, e-learning platforms, and online resources to enhance and deliver educational content to students.

1.8 Organization of the Thesis

This thesis is structured to facilitate a coherent exploration of the research objectives. Chapter 2 provides an extensive literature review, offering a foundation for understanding the complexities of digital learning, data architecture, and cybersecurity in various educational settings. Chapter 3 details the research methodology, outlining the design, participants, and data analysis procedures. Subsequent chapters delve into survey findings, comparative analysis, security model design, Secure Grade Distribution Scheme development, technological recommendations, and a proof of concept. The final chapters offer a comprehensive discussion of the implications, contributions, limitations, and future recommendations arising from this research.

This thesis is structured to facilitate a coherent exploration of the research objectives. Chapter 2 provides an extensive literature review, offering a foundation for understanding the complexities of digital learning, data architecture, and cybersecurity in various educational settings. Chapter 3 details the research methodology, outlining the design, participants, and data analysis procedures. Subsequent chapters delve into survey findings, comparative analysis, security model design, Secure Grade Distribution Scheme development, technological recommendations, and a proof of concept. The final chapters offer a comprehensive discussion of the implications, contributions, limitations, and future recommendations arising from this research.

CHAPTER TWO: LITERATURE REVIEW

2.1 Preamble

The literature review in this chapter unfolds as a comprehensive exploration of the foundational concepts essential to understanding the intricate interplay of digital learning, data architectures, security models, and grade distribution schemes within higher education. By delving into the evolving landscape of these domains, this chapter seeks to unearth critical insights, identify existing gaps, and establish a theoretical framework that aligns with the research objectives.

Education, particularly in higher institutions, has experienced a profound shift with the integration of digital technologies. To comprehend this transformative journey, the literature review embarks on an exploration of digital learning in higher education. The evolution, trends, challenges, and opportunities within this dynamic realm lay the groundwork for understanding the broader context in which data architectures, security models, and grade distribution schemes operate.

Moving seamlessly into the realm of data architectures, the review scrutinizes the existing models adopted by universities. This includes a nuanced examination of centralized and decentralized architectures, distributed databases, and data warehouses. Simultaneously, a spotlight is cast on the cybersecurity challenges inherent in these architectures, unravelling the intricacies of safeguarding digital assets against an evolving landscape of threats.

The subsequent sections navigate through security models in digital learning environments, surveying previous approaches and discerning their limitations. It is within this critical evaluation that the chapter aims to identify areas ripe for innovation and enhancement, contributing to the development of a robust security model. The exploration concludes with an in-depth analysis of grade distribution schemes, evaluating current practices and probing into the security concerns that underscore the dissemination of academic assessments in digital settings.

This chapter not only reviews relevant literature but also synthesizes and analyses it, providing a foundation for the subsequent research methodology and data analysis. As the

narrative unfolds, the reader is invited to traverse the terrain of digital transformation in higher education, where the amalgamation of technology, security, and academic assessment converges, setting the stage for the novel contributions and insights that this research endeavours to unveil.

2.2 Theoretical Framework

2.2.1 Digital Learning in Higher Education

The landscape of higher education has undergone a profound transformation with the integration of digital learning methodologies. This section explores the evolution, trends, challenges, and opportunities that define the intricate relationship between technology and education in the higher academic sphere.

2.2.1.1 Evolution and Trends

The evolution of digital learning in higher education represents a transformative journey from its embryonic stages to the sophisticated models witnessed today. Early experiments with computer-assisted instruction set the foundation for the development of more advanced Learning Management Systems (LMS) and collaborative online platforms (Bervell & Umar, 2017). The initial focus on digitizing content gradually shifted towards more interactive and learner-centric approaches, emphasizing personalized educational experiences.

Historically, the evolution has been marked by a move from static, one-size-fits-all educational content to dynamic, adaptive learning technologies. Artificial Intelligence (AI) has played a crucial role in this evolution, offering the capability to analyze vast amounts of student data. AI-driven tools, such as personalized learning platforms, can tailor educational content to individual learning styles and pace (Bezovski & Poorani, 2016). This evolution represents a paradigm shift towards a more individualized and responsive educational environment.

Digital learning's roots can be traced back to early experiments with programmed instruction, with B.F. Skinner's teaching machine being a noteworthy example (Skinner, 1957). However, it was the advent of the internet and the subsequent development of

online learning platforms that propelled digital learning into mainstream education. The introduction of Learning Management Systems (LMS), such as Blackboard and Moodle, marked a significant leap, providing a centralized platform for course management and content delivery (Allen & Seaman, 2017). This phase laid the groundwork for the subsequent evolution by bringing education into the digital realm.

As the 21st century unfolded, digital learning witnessed a shift towards more dynamic and interactive models. The concept of Massive Open Online Courses (MOOCs) emerged, offering scalable and accessible educational content to a global audience (Daniel, 2012) MOOCs exemplify the democratization of education, breaking down geographical barriers and providing learners with the flexibility to engage with course materials at their own pace. This period saw a transition from traditional, instructor-centred models to more learner-centric approaches, reflecting a growing understanding of the diverse needs and preferences of students in a digital age.

2.2.1.2 Trends in Digital Learning

The trends shaping digital learning in higher education are dynamic, responding to the evolving needs of learners and advancements in technology. Mobile learning, facilitated by the ubiquity of smartphones and tablets, has become a prevalent trend, providing learners with the flexibility to access educational content anytime and anywhere (Ferguson et al., 2019) This trend aligns with the societal shift towards a mobile-centric lifestyle, making education more accessible and convenient.

Moreover, the integration of immersive technologies has become a defining trend in digital learning. Virtual Reality (VR) and Augmented Reality (AR) are reshaping traditional learning environments. VR immerses learners in computer-generated scenarios, facilitating experiential learning in fields like science and medicine (Maurice et al., 2014). AR overlays digital content onto the real world, creating interactive learning experiences. These technologies contribute to a more engaging and interactive educational experience.

Simultaneously, social learning platforms have gained prominence as a trend in digital education. Platforms like Edmodo and Schoology provide spaces for collaborative learning, enabling students to interact, share resources, and engage in discussions beyond

the confines of the physical classroom (Dabbagh & Kitsantas, 2012). Social learning leverages the power of online communities, fostering a sense of belonging and facilitating peer-to-peer learning.

The recent global response to the COVID-19 pandemic has accelerated existing trends. The widespread adoption of remote and online learning has become a dominant trend, emphasizing the importance of digital learning in ensuring continuity during times of disruption (Hodges et al., 2020). This trend has showcased the adaptability and resilience of digital learning models in the face of unprecedented challenges.

2.2.1.3 Challenges and Opportunities

Digital learning, while offering transformative possibilities, is not without its challenges. One significant obstacle is the existence of a digital divide, representing disparities in access to technology and the internet among different demographic groups (Warschauer & Matuchniak, 2010). Bridging this gap is crucial for achieving inclusive education, as learners without adequate access may be excluded from the benefits of digital learning. Initiatives to address the digital divide must be comprehensive, considering infrastructural, economic, and educational aspects.

Ensuring the quality of online education is another pressing challenge. The shift to virtual classrooms necessitates careful considerations regarding the effectiveness of digital learning experiences (Selwyn, 2016). Developing robust strategies for designing and delivering high-quality content, assessments, and interactive elements is essential. Moreover, issues related to digital literacy and the ability of learners to navigate online platforms can impact the overall quality of the learning experience (Mackey & Jacobson, 2011).

Teacher training stands out as a critical challenge in the digital learning landscape. Educators need to be equipped with the skills and knowledge to effectively leverage digital tools and technologies for teaching and learning. This involves not only technical proficiency but also pedagogical strategies that integrate digital resources seamlessly into the curriculum (Ertmer et al., 2012). Continuous professional development becomes imperative to ensure educators remain adept in the rapidly evolving digital landscape.

The rapid pace of technological advancements compounds the challenges in digital learning. Continuous adaptation is required not only from educators but also from institutions and policymakers to keep abreast of the latest innovations and best practices (Bates, 2019). Ensuring that educational institutions have the capacity and flexibility to integrate emerging technologies responsibly is a multifaceted challenge that requires strategic planning and collaboration.

Issues related to student engagement in virtual classrooms also merit attention. Maintaining a sense of connection and interaction in digital environments poses unique challenges. Strategies to enhance student engagement need to be explored, encompassing both synchronous and asynchronous elements of digital learning (Crompton, 2013). Balancing flexibility with structured engagement becomes crucial to foster a sense of community among learners.

Moreover, the diversity of digital learning tools and platforms introduces challenges related to standardization and interoperability. Integrating various technologies seamlessly into the learning environment can be complex and may require consistent standards to ensure a cohesive experience for both educators and learners (Bates, 2019). Achieving interoperability can enhance the efficiency and effectiveness of digital learning ecosystems.

2.2.1.4 Opportunities in Digital Learning:

Despite these challenges, digital learning presents unprecedented opportunities for the higher education sector. The recent global response to the COVID-19 pandemic showcased the resilience and adaptability of digital learning models (Hodges et al., 2020). Opportunities include expanded access to education, as learners can participate in courses and programs from anywhere in the world. The digital landscape also fosters collaboration and engagement through various online platforms, allowing for interactive and dynamic learning experiences.

Innovation in pedagogical approaches is a significant opportunity presented by digital learning. The flexibility of digital tools enables educators to explore new ways of delivering content, fostering critical thinking, and promoting active learning (Bates, 2019).

Virtual laboratories, simulations, and gamified elements provide avenues for experiential learning, enhancing the depth and breadth of educational experiences(Maurice et al., 2014). This opens possibilities for educators to tailor instruction to individual learning styles, catering to diverse student needs.

Additionally, digital learning facilitates the creation of inclusive learning environments, accommodating diverse learning styles and preferences. Customizable learning paths, adaptive assessments, and personalized feedback contribute to an individualized learning experience (Means & Neisler, 2021). This inclusivity extends beyond geographical boundaries, offering educational opportunities to learners who might face constraints in traditional settings.

The widespread use of analytics in digital learning platforms presents an opportunity to gather valuable insights into student performance and engagement. Learning analytics can inform educators about effective teaching strategies, areas where students may need additional support, and the overall effectiveness of the learning materials (Siemens & Long, 2011). Harnessing the power of data-driven insights allows for continuous improvement in educational practices.

Moreover, digital learning offers the potential for lifelong learning and continuous skill development. Online courses, micro-credentials, and digital badges provide avenues for learners to acquire new skills and knowledge throughout their lives, fostering a culture of continuous learning (Hodges et al., 2020). This aligns with the evolving needs of the workforce, where adaptability and upskilling are increasingly crucial.

2.3 Data Architectures in Higher Education

The orchestration of data architectures assumes a pivotal role in shaping the digital learning landscape within higher education institutions. This section intricately explores the diverse structures and frameworks that underpin the storage, management, and utilization of data in the complex realm of educational settings.

2.3.1 Existing Models

Within the realm of higher education, a rich tapestry of data architectures has unfolded, each offering a distinctive approach to addressing the intricate needs of digital learning environments. The centralized data architecture stands as a prominent model, characterized by a singular repository serving as the focal point for all educational data (Boh Podgornik et al., 2016). Praised for its simplicity in management and uniform data access, this model contributes to fostering a cohesive learning environment. However, potential challenges in scalability and adaptability may emerge, especially when confronted with the dynamic landscape of evolving digital learning technologies.

In contrast, federated data architectures introduce a decentralized paradigm, distributing data across multiple repositories maintained by distinct departments or units within a university (Guo & Zeng, 2020). This approach fosters autonomy in data governance but may encounter challenges related to data consistency and interoperability between disparate systems.

The advent of cloud-based data architectures has ushered in a new era, leveraging cloud computing services for the storage and processing of educational data (Al-Malah et al., 2021). Renowned for scalability, flexibility, and accessibility, cloud-based solutions empower institutions to swiftly adapt to changing demands. However, careful consideration is essential regarding data security, privacy, and the potential implications of relying on external service providers.

Graph-based data architectures have gained prominence for their adeptness in representing intricate relationships within educational datasets (Nakagawa et al., 2019). Graph databases excel in capturing connections between various data points, offering a nuanced understanding of student interactions, course dependencies, and institutional dynamics. This model aligns seamlessly with the emphasis on personalized and interconnected educational experiences in digital learning environments.

Hybrid data architectures have emerged as a pragmatic approach, combining elements of centralized, federated, and cloud-based models (Guo & Zeng, 2020). This versatile

approach allows institutions to tailor their data architecture to specific needs, striking a harmonious balance between standardization and customization.

2.3.2 Cybersecurity Challenges

In the intricate landscape of data architectures within higher education, cybersecurity emerges as a paramount concern, shaping the resilience and integrity of digital learning environments. This section delves into the multifaceted cybersecurity challenges faced by existing data architectures, emphasizing the importance of safeguarding educational data.

Cybersecurity challenges within higher education data architectures are diverse and dynamic, demanding vigilant attention to mitigate potential risks. One significant challenge arises from the increasing sophistication of cyber threats targeting educational institutions (Joksimović et al., 2019). Malicious actors often exploit vulnerabilities within data architectures to gain unauthorized access, compromise sensitive information, or disrupt essential educational services. The rapid evolution of these threats necessitates continuous adaptation and proactive measures to ensure the security of educational data.

Data privacy concerns constitute a critical facet of cybersecurity challenges in higher education data architectures (Carvalho Ota et al., 2020). As educational institutions amass vast amounts of student and faculty data, ensuring compliance with data protection regulations becomes paramount. Unauthorized access, data breaches, or inadvertent disclosures can result in severe consequences, not only compromising individuals' privacy but also undermining institutional trust and reputation.

The interconnectedness of data architectures in digital learning environments amplifies the challenge of securing sensitive information. The sharing and exchange of data between various components of the educational ecosystem create potential vulnerabilities that malicious actors may exploit (Djeki et al., 2023). This interconnectedness necessitates a holistic approach to cybersecurity, addressing vulnerabilities at the system, network, and application levels.

Moreover, the increasing reliance on cloud-based data architectures introduces new cybersecurity challenges. While cloud solutions offer scalability and flexibility, they also

expose educational institutions to risks associated with third-party service providers (Al-Malah et al., 2021). Data breaches, service disruptions, or inadequate security measures implemented by cloud providers can have profound implications on the confidentiality and availability of educational data.

Insider threats pose a significant cybersecurity challenge, highlighting the importance of robust internal controls and user awareness (Bhatia & Maitra, 2018). Individuals within educational institutions, intentionally or unintentionally, may compromise data security. Mitigating insider threats requires a combination of technical controls, employee training, and effective monitoring to detect and respond to suspicious activities.

The integration of emerging technologies, such as Internet of Things (IoT) devices and Artificial Intelligence (AI), into educational data architectures introduces additional cybersecurity challenges (Li et al., 2019). Ensuring the security of these technologies and their seamless integration with existing data architectures is crucial to prevent potential exploitation by cyber adversaries.

2.4 Security Models in Digital Learning Environments

The secure operation of digital learning environments relies on resilient security models that protect sensitive data and uphold the integrity of educational processes. This section delves into the evolution of security models within the realm of digital learning, with a specific focus on exploring vulnerabilities addressed by previous approaches to fortify the resilience of these environments.

2.4.1 Previous Approaches

Security models in digital learning environments have undergone transformations in response to dynamic cyber threats and the evolving landscape of educational technologies.

Previous approaches predominantly emphasized perimeter-based security, concentrating on fortifying external boundaries to thwart unauthorized access (Bhatia & Maitra, 2018). Utilizing firewalls, intrusion detection systems, and secure network configurations, this approach sought to create a secure perimeter around digital learning systems. However, it

grappled with limitations, particularly in addressing internal threats and countering sophisticated cyber-attacks that could circumvent traditional perimeter defences.

Authentication and access control mechanisms constituted critical components of earlier security models in digital learning (Aissaoui & Azizi, 2017). Implementing user authentication through passwords, multi-factor authentication, and role-based access control aimed to ensure that only authorized individuals could access sensitive educational data and resources. Despite their effectiveness to a certain extent, these mechanisms faced challenges related to password vulnerabilities, user compliance, and the dynamic nature of user roles within educational institutions.

Cryptography played a fundamental role in previous security models, focusing on encrypting data to safeguard its confidentiality during both transmission and storage (Alassery, 2021). Widely used protocols such as Secure Sockets Layer (SSL) and Transport Layer Security (TLS) were employed to establish secure communication channels. However, challenges surfaced in maintaining cryptographic protocols up-to-date, securely managing cryptographic keys, and addressing vulnerabilities associated with encryption algorithms.

The shift towards a more holistic and adaptive security posture led to the development of risk-based security models in digital learning environments (Mihailescu et al., 2020). Rather than relying solely on predetermined security measures, risk-based models assess contextual factors, user behavior, and emerging threats to dynamically adjust security controls. This approach acknowledges the dynamic nature of digital learning environments and aims to strike a balance between security, usability, and adaptability.

Furthermore, the Zero Trust Security Model gained prominence as a response to the limitations of perimeter-based approaches (Arabi, 2021). In the Zero Trust model, trust is never assumed, and verification is required from anyone attempting to access resources, even within the internal network. This approach minimizes the potential impact of insider threats, operating on the assumption that the internal network is as untrusted as external networks.

2.4.2 Limitations and Gaps

While previous security approaches in digital learning environments have made strides in fortifying the integrity of educational processes, they are not without their limitations and identifiable gaps. This section scrutinizes the vulnerabilities and shortcomings inherent in these models, shedding light on areas that demand further attention and innovation.

One significant limitation lies in the reliance on perimeter-based security models, which, while providing a degree of protection, struggle to address internal threats effectively (Bhatia & Maitra, 2018). The traditional emphasis on securing external boundaries often leaves digital learning environments susceptible to insider threats and sophisticated attacks that navigate through the established defences. The permeability of these perimeters poses a persistent challenge in safeguarding against threats originating within the educational ecosystem.

Authentication and access control mechanisms, while essential, grapple with vulnerabilities tied to human behaviour and compliance (Aissaoui & Azizi, 2017). Password-based authentication remains susceptible to issues such as weak password practices, password reuse, and the challenge of enforcing robust password policies across diverse user groups. Additionally, as user roles evolve within educational institutions, maintaining an accurate and dynamic representation of access privileges becomes an ongoing challenge, leading to potential security gaps.

Cryptography, a stalwart in data protection, encounters limitations in managing the complexity of cryptographic protocols and keys (Alassery, 2021). Keeping cryptographic protocols up to date with emerging standards and mitigating vulnerabilities associated with encryption algorithms necessitate continuous attention. The secure management of cryptographic keys, crucial for maintaining the confidentiality of data, demands robust practices to prevent unauthorized access and potential compromise.

Risk-based security models, while providing adaptability, introduce complexities in assessing and responding to dynamic contextual factors (Mihailescu et al., 2020). The effectiveness of these models hinges on accurate risk assessments, which can be challenging given the evolving nature of cyber threats and the intricate interplay of factors

influencing security postures. Striking the right balance between security measures, usability, and adaptability remains a delicate challenge in the implementation of risk-based models.

The Zero Trust Security Model, although a paradigm shift, presents challenges in practical implementation and cultural adaptation within educational institutions (Arabi, 2021).. Overcoming ingrained trust assumptions and seamlessly integrating the Zero Trust approach into existing digital learning environments requires comprehensive planning and organizational readiness.

Moreover, the increasing integration of emerging technologies, such as the Internet of Things (IoT) and Artificial Intelligence (AI), poses new challenges in terms of security (Lu & Da Xu, 2018). Ensuring the security of these technologies and their harmonious coexistence with existing security models demand continuous vigilance and adaptation.

2.4.3 Review of Related Work

In the exploration of security models within digital learning environments, an exhaustive review of related literature has been conducted, encompassing the below notable works that provide diverse insights into the development, challenges, and advancements in this critical domain.

Security models and frameworks are important in both e-learning and university systems to ensure the protection of sensitive information and prevent cyber threats. In the context of e-learning, the use of security and cyber security countermeasures has been found to have a significant impact on students' frequent use and participation in the system (Kale et al., 2023). Additionally, student feedback and communication about their e-learning experience can help address security concerns and increase participation(Al-Sherideh et al., 2023). In the context of university automation systems, information security frameworks have been proposed to ensure the overall safety of the system(Hasan et al., 2022). These frameworks aim to protect the information of different confidentiality levels and ensure sustainable information security (Ramanauskaitė et al., 2021). Furthermore, a security concept model has been developed for distance learning, consisting of security assurance,

users, and organizational processes, with technical measures provided at the system administrator level.

In conclusion, these works collectively contribute a nuanced understanding of security models in digital learning environments, each offering valuable insights, methodologies, and recommendations. While presenting significant advancements, these studies also acknowledge inherent limitations, underscoring the evolving nature of research in this dynamic field (Table 1).

Table 1 Summary of Related Work 1

Reference	Objectives	Method	Contribution	Limitation
(Kale et al., 2023)	The provided paper proposes an information security framework for a university automation system	Blog crawling and Traditional document search	• Identification of shared resources in university automation system. • Proposal of an information security framework for overall system safety.	The research did not provide secure data storage.
(Al-Sherideh et al., 2023)	The paper focuses on developing e-learning security model, the impact of security measures on students' academic achievements.	Collecting information on e-learning security measures and students' perceptions. Designing a questionnaire and selecting the right sample of respondents.	Assessing the impact of security measures on students' academic achievements. Development of a security model to detect cyberattacks.	The model only detect attack does not prevent it
(Hasan et al., 2022)	The provided paper proposes an information security framework for a university automation system.	Identification of shared resources in university automation system Proposal of information security framework for overall system safety	Identification of shared resources in university automation system. Proposal of an information security framework for overall system safety.	The model did not provide data security for the university

(Jusas et al., 2022)	The paper provides a framework for implementing security systems in e-learning environment	Explaining and establishing frameworks for implementing security systems and Modelling threats posed by a malicious hacker	Identification of shared resources in university automation system. Proposal of an information security framework for overall system safety.	Lack of proper mitigation technique
(Ramanauskaitė et al., 2021)	The paper proposes a security level estimation model for educational organizations	The article developed a security level estimation model for educational organizations.	Proposal of a security level estimation model for educational organizations. Validation of the proposed model through use case analysis and expert evaluation.	Lack of security level modelling for educational organizations.
(Modesti, 2020)	The provided paper proposes an information security framework for a university automation system.	Integration of formal methods for security research into teaching practice and adoption of a conceptual model aligned with high-level representation of cryptographic and communication primitives	Identification of shared resources in university automation system. Proposal of an information security framework for overall system safety.	No efficient method of preventing identified cyberattacks

2.5 Grade Distribution Schemes

Grade distribution schemes play a pivotal role in assessing and communicating student performance within educational institutions. This section examines the current practices employed in grade distribution schemes, shedding light on the methodologies and frameworks used to evaluate and disseminate academic achievements.

2.5.1 Current Practices

In contemporary educational landscapes, various institutions employ diverse grade distribution schemes to represent student accomplishments fairly and transparently. The

prevalent approach involves the utilization of letter grades, ranging from A to F, each corresponding to a specific level of achievement. This traditional system allows for a straightforward classification of performance, with 'A' typically denoting excellent performance and 'F' indicating failure.

Percentage-based grading is another widely adopted practice wherein students receive a numerical score reflective of their performance relative to the maximum achievable score. This system provides a granular representation of achievement, allowing for subtle distinctions in performance. However, challenges may arise in cases of standardized testing, where a fixed scale may not align with the difficulty level of different assessments.

Some institutions embrace a pass/fail system, simplifying the evaluation process by categorizing students as either passing or failing without assigning specific grades. This approach aims to reduce academic stress and foster a focus on learning rather than grades. However, it may lack the granularity necessary for detailed academic assessments.

In recent years, competency-based grading has gained traction, emphasizing the mastery of specific skills and knowledge rather than traditional letter or percentage grades. This approach is particularly prevalent in competency-driven programs and is designed to provide a more nuanced understanding of a student's capabilities.

Additionally, narrative evaluations offer a qualitative alternative, providing detailed written assessments of a student's performance. This personalized approach allows instructors to offer tailored feedback, emphasizing strengths and areas for improvement. While narrative evaluations can offer a comprehensive view of a student's progress, they may lack the standardization and comparability of more quantitative systems.

The shift towards digital learning and assessment tools has prompted the exploration of automated grading systems. These systems utilize algorithms to assess assignments, quizzes, and exams, providing quick feedback to students. While efficient, concerns persist regarding the potential limitations in capturing the nuanced aspects of student performance that may be evident in qualitative assessments.

2.5.2 Security Concerns

As educational institutions continue to embrace digital platforms for grade distribution, a set of security concerns emerges, demanding vigilant attention to safeguard the integrity and confidentiality of academic assessments. This section scrutinizes the potential security challenges associated with digital grade distribution schemes, emphasizing the need for robust measures to mitigate risks.

One primary security concern revolves around data breaches and unauthorized access to grade databases. The digital storage and transmission of sensitive student information necessitate stringent measures to prevent malicious actors from gaining unauthorized access. Encryption protocols and secure authentication mechanisms become imperative to safeguard against unauthorized intrusion and data tampering.

The potential manipulation of grades poses another significant security threat. Malicious actors may attempt to alter grades either for personal gain or to create disruptions within the educational system. Ensuring the integrity of the grade distribution system requires implementing measures such as digital signatures and audit trails to detect and prevent unauthorized grade changes.

Phishing attacks targeting students or faculty members represent a tangible threat to the confidentiality of grade information. Cybercriminals may employ deceptive tactics to trick individuals into divulging login credentials, enabling unauthorized access to the grade distribution platform. Educational institutions must prioritize cybersecurity awareness programs to mitigate the risks associated with social engineering attacks.

The reliance on third-party grading platforms introduces concerns related to the security practices of external service providers. Educational institutions must rigorously assess and monitor the security measures implemented by these platforms to ensure compliance with data protection regulations and prevent potential vulnerabilities that may compromise student data.

The potential for distributed denial-of-service (DDoS) attacks on grade distribution systems poses a disruptive threat. An orchestrated DDoS attack could overwhelm the system,

rendering it temporarily inaccessible and causing disruptions during critical grading periods. Implementing robust network infrastructure and DDoS mitigation strategies becomes crucial to maintain system availability and resilience.

Moreover, the vulnerability of automated grading systems to algorithmic biases and errors raises ethical and security concerns. Biases in algorithms may disproportionately impact certain student groups, leading to unfair assessments. Ensuring transparency in algorithmic decision-making and regularly auditing automated grading systems can address these ethical and security considerations.

2.5.3 Review of Related Work

In this (Plyer et al., 2022), the authors created a unique method for grading chemistry examinations in Moodle. Their plugin can properly grade chemistry tests, and the mark is safely stored in Moodle. Other authors (Pérez et al., 2017) suggested a method for detecting any modification of Moodle grades and alerting the users in charge to maintain the grades' security. The article focuses on SQL injection, a code injection attack that targets data-driven systems that introduce malicious SQL statements into a field for execution. The suggested solution may detect student grade changes and inform the instructor. It was created using PHP. However, the research was limited to detecting SQL injection and did not include prevention methods. (Abdelsalam et al., 2023) In their study aimed to enhance the security of Moodle's grade distribution system. They proposed a new encryption scheme to protect grade data during transmission. The research introduces cryptographic techniques to safeguard sensitive information. (Cyoy, 2022) focuses on implementing two-factor authentication in Moodle to ensure secure access to grade-related information. The study explores methods to add an extra layer of protection to prevent unauthorized access to student grades. (Korać et al., 2022b) investigated the vulnerabilities of Moodle's gradebook and proposed strategies to strengthen its security. The study delves into potential threats and provides recommendations to address weaknesses in the Moodle platform's grade management system. (Elmaghrabi & Eljack, 2019) provided a comprehensive review of existing security measures in Moodle's grade distribution is presented in this research. The authors analyze the strengths and weaknesses of current methods and suggest improvements to enhance overall system security (Table 2).

Table 2. Summary of Related Work 2

Reference	Objective	Contribution	Limitations
(Pérez et al., 2017)	The objective of the research is to prevent changes in student grades in the Moodle platform	The study suggested a solution that will detect any change in a student's status and inform the instructor of it.	The research has limitations in detecting SQL injection and did not include prevention methods. The research only provides means of detecting changes in grades, not preventing them.
(Plyer et al., 2022)	Providing a new grading method for chemistry exams and safe grade storage inside the Moodle platform is the primary objective of the work.	The study developed and installed a Moodle plugin for grading chemistry examinations.	The research did not develop any security technique for preventing data breaches in grades in the Moodle platform.
(Abdelsalam et al., 2023)	Enhancing the security of Moodle's grade distribution system using a new encryption scheme	Introduction of cryptographic techniques to safeguard sensitive information	The research did not provide a secured way of sharing grades with staff and students
(Cyoy, 2022)	Implementing two-factor authentication in Moodle to secure access to grade-related information	Exploration of methods to add an extra layer of protection	The research did not provide encryption for student's grades.
(Korać et al., 2022b)	Investigating vulnerabilities in Moodle's gradebook and proposing strategies for improvement	In-depth analysis of potential threats and recommendations	Limited information on the practical implementation of suggested strategies
(Elmaghrabi & Eljack, 2019)	Reviewing existing security measures in Moodle's grade distribution	Analysis of strengths and weaknesses, suggestions for improvements	Lack of empirical testing for proposed enhancements

CHAPTER THREE: METHODOLOGY

3.1 Preamble

The research design serves as the architectural framework that guides the systematic inquiry into the multifaceted aspects of a secure data-centric model for digital learning in higher education, particularly within the context of West African universities. This section delineates the blueprint and methodology employed to rigorously explore the existing data architectures, cybersecurity challenges, and grade distribution schemes prevalent in the evolving landscape of digital education.

A judicious selection of a mixed-methods research design is deemed imperative for its capacity to amalgamate the strengths of both qualitative and quantitative methodologies. This combination facilitates a comprehensive and nuanced investigation into the intricacies of data security and digital learning. The qualitative dimension unfolds through in-depth interviews, engaging key stakeholders to extract rich narratives and perspectives. Concurrently, the quantitative facet leverages surveys and statistical analyses to quantify prevailing trends, assess the efficacy of security measures, and gauge satisfaction levels with current grade distribution systems.

The integrity and robustness of the research endeavour hinge on the meticulous design and calibration of data collection instruments. Interview guides, tailored to the intricacies of data architectures and digital learning, are poised to elicit profound insights. The structured questionnaire for surveys draws on validated measures to ensure the reliability and validity of the gathered quantitative data.

A purposive sampling strategy has been strategically devised to assemble a diverse cohort of participants, encompassing educators, IT administrators, and students from West African universities. This approach seeks to capture a spectrum of perspectives reflective of the regional nuances and challenges inherent in the subject matter.

As ethical considerations stand as a cornerstone of responsible research, this section underscores the commitment to upholding ethical standards. Rigorous adherence to participant confidentiality, informed consent, and responsible data handling are paramount.

The research protocol will be subjected to scrutiny and approval by the pertinent institutional review board, affirming the ethical rigor of the study.

3.2 Problem Formulation

The formulation of the research problem serves as the compass directing the inquiry into the secure data-centric model for digital learning in higher education, particularly within the dynamic context of West African universities. This section meticulously defines and articulates the challenges and gaps that motivate the research, providing a clear trajectory for the investigation.

3.2.1 Data Architectures

In the intricate landscape of higher education, the efficacy and security of data architectures stand as pivotal determinants of the digital learning experience. This subsection delves into the nuanced realm of data architectures within West African universities, aiming to meticulously identify vulnerabilities that may compromise the integrity, accessibility, and confidentiality of academic information.

3.2.2 Systemic Analysis of Existing Data Architectures

The foundation of any digital learning environment lies in its data architecture. Through a comprehensive survey, this research endeavours to conduct a systemic analysis of the prevailing data architectures across West African universities. This includes an exploration of the infrastructure, databases, and data storage mechanisms employed, with a keen focus on understanding their design principles and implementation intricacies.

I. Examination of Cybersecurity Incidents

A critical dimension of identifying vulnerabilities is an in-depth examination of past cybersecurity incidents. By scrutinizing the historical landscape, this research aims to catalogue and analyse instances of cyber-attacks faced by West African universities. Understanding the modus operandi of these incidents is imperative for uncovering potential weak points within data architectures and formulating targeted strategies for fortification.

II. Evaluation of Countermeasures Implemented

In tandem with identifying vulnerabilities, an assessment of the countermeasures implemented by universities becomes paramount. This research will investigate the proactive measures taken by institutions to mitigate and respond to cybersecurity challenges. The evaluation extends beyond technological solutions to encompass policies, training programs, and organizational protocols designed to bolster the resilience of data architectures.

III. Regional and Institutional Variances

Recognizing the diversity across West African universities, this research will be attentive to regional and institutional variances in data architectures. Understanding the unique challenges faced by different institutions ensures that interventions and recommendations are contextually relevant. Factors such as infrastructure limitations, resource availability, and regional threat landscapes will be considered in this nuanced analysis.

IV. Integration of Lessons Learned from the COVID-19 Pandemic

The paradigm shift in educational practices catalyzed by the COVID-19 pandemic has underscored the significance of robust data architectures. Lessons learned from this transformative period will be integrated into the identification of vulnerabilities, considering the specific challenges and adaptations made by West African universities during this global crisis.

3.2.3 Cybersecurity Challenges during the Pandemic

The unprecedented shift to remote and digital learning catalyzed by the COVID-19 pandemic has brought forth a myriad of cybersecurity challenges within the realm of higher education. This subsection delves into the multifaceted landscape of cybersecurity challenges faced by West African universities during the pandemic, aiming to unravel the intricacies of digital vulnerabilities and potential threats to academic data.

I. Surge in Phishing and Social Engineering Attacks

With the surge in digital communication channels, the pandemic ushered in an alarming increase in phishing and social engineering attacks. Malicious actors

exploited the uncertainties and urgency surrounding the pandemic, targeting students, faculty, and administrators. This research will dissect the methodologies employed in these attacks, shedding light on the vulnerabilities exposed within the digital communication infrastructure of universities.

II. Scalability Issues and Technological Gaps

The abrupt transition to digital learning exposed scalability issues and technological gaps in existing cybersecurity infrastructures. West African universities faced challenges in scaling up their security measures to accommodate the sudden influx of online activities. This research seeks to identify the specific technological gaps and scalability bottlenecks that impeded effective cybersecurity responses during the pandemic.

III. Data Privacy Concerns in Remote Learning Environments

Remote learning, while crucial for continuity, introduced concerns regarding data privacy. The transition to online platforms for lectures, examinations, and collaborative projects raised questions about the protection of sensitive student information. This research will investigate the extent of data privacy concerns, examining the adequacy of existing measures and proposing strategies to enhance the safeguarding of student data.

IV. Adapting to Evolving Cyber Threats

The dynamic nature of cyber threats demands continual adaptation from educational institutions. This research aims to analyze how West African universities adapted to evolving cyber threats during the pandemic. It will explore the agility of existing cybersecurity frameworks, the integration of threat intelligence, and the responsiveness of incident response mechanisms.

V. Impact of Increased Network Traffic

The surge in online activities during the pandemic led to a substantial increase in network traffic within educational institutions. This subsection will explore the impact of heightened network usage on cybersecurity, assessing the resilience of network infrastructures, potential bottlenecks, and strategies employed to maintain network security while accommodating increased demand.

VI. Collaboration and Communication Security

As collaboration tools became integral to remote learning, ensuring the security of communication channels and collaborative platforms became paramount.

This research will scrutinize the cybersecurity challenges associated with the adoption of virtual communication tools, emphasizing the need for secure channels for academic discourse and collaboration.

3.2.4 Grade Distribution Scheme Vulnerabilities

The distribution of grades is a cornerstone of academic evaluation, and the transition to digital learning platforms has brought forth a unique set of vulnerabilities within grade distribution schemes. This subsection meticulously examines the vulnerabilities inherent in the systems responsible for disseminating student grades in West African universities, aiming to fortify the confidentiality, accuracy, and overall integrity of the grading process.

I. Integrity and Authenticity of Digital Grade Repositories

The digitalization of grade repositories introduces challenges related to the integrity and authenticity of academic records. This research will scrutinize the vulnerabilities associated with the storage and management of digital grades, including the risk of unauthorized access, tampering, or manipulation. Strategies for ensuring the trustworthiness of these repositories will be explored.

II. Potential Exploitation of Online Examination Systems

With the surge in online examinations, concerns arise regarding the potential exploitation of these systems. This research delves into the vulnerabilities associated with digital examination platforms, including the risk of cheating, impersonation, or manipulation of examination results. Strategies for enhancing the security of online examination systems will be considered.

III. Privacy Concerns in Digital Grade Transmission

The transmission of grades in digital formats raises privacy concerns, especially regarding the secure and confidential communication of academic results. This subsection explores vulnerabilities in the transmission process, including the risk of interception or unauthorized access. Recommendations for ensuring encrypted and secure grade transmission will be proposed.

IV. Accessibility and Inclusivity Challenges

While digital grade distribution offers convenience, it may inadvertently introduce accessibility challenges. This research examines vulnerabilities related to the

inclusivity of digital grade distribution systems, considering factors such as internet access disparities, technological barriers, and the potential exclusion of certain student groups. Strategies for fostering inclusivity will be addressed.

V. Technological Infrastructure Limitations

The effectiveness of digital grade distribution is contingent on the technological infrastructure supporting it. This research investigates vulnerabilities arising from technological limitations, such as server downtimes, bandwidth constraints, or compatibility issues. Recommendations for bolstering technological resilience in grade distribution schemes will be explored.

VI. Risk of Algorithmic Biases in Automated Grading

The adoption of automated grading systems introduces the risk of algorithmic biases. This subsection explores vulnerabilities related to the fairness and impartiality of automated grading algorithms, considering potential disparities in grading outcomes based on demographic or contextual factors. Strategies for mitigating algorithmic biases will be scrutinized.

3.2.5 Impact on Digital Learning Experience

The dynamic integration of digital technologies into higher education has redefined the learning experience, yet this transformation is not without challenges. This subsection delves into the multifaceted impacts on the digital learning experience within the context of West African universities. By examining both positive and negative implications, the research seeks to provide a holistic understanding of the consequences of the digital shift on students, educators, and the academic ecosystem.

I. Enhanced Accessibility and Flexibility

One of the positive impacts of the digital learning experience is the enhanced accessibility and flexibility it offers. Students can engage with educational materials and participate in classes from virtually anywhere. This subsection will explore how these advantages have positively influenced the learning experience, enabling more inclusive and flexible educational practices.

II. Challenges in Technological Adaptation

Conversely, the rapid shift to digital learning has brought about challenges in technological adaptation. This research will scrutinize how students and educators

navigate the learning curve associated with digital tools, examining potential barriers and disparities in technological proficiency that may affect the overall learning experience.

III. Interactive Learning Opportunities

Digital learning platforms often facilitate interactive learning opportunities through forums, collaborative projects, and virtual discussions. This subsection aims to highlight the positive impact of these interactive elements on student engagement, knowledge retention, and the overall quality of the learning experience.

IV. Social Isolation and Reduced Engagement

On the flip side, the digital learning experience has been associated with social isolation and reduced engagement for some students. This research will investigate the impact of virtual learning environments on social interactions, community-building, and the sense of belonging within the academic community.

V. Adaptation of Pedagogical Approaches

The adoption of digital tools has prompted a reconsideration of pedagogical approaches. This subsection will delve into how educators have adapted their teaching methods to the digital landscape, exploring innovations in online teaching, assessment strategies, and the integration of multimedia resources.

VI. Digital Fatigue and Cognitive Overload

Digital learning, if not well-managed, can contribute to digital fatigue and cognitive overload. This research aims to understand the negative impact of prolonged screen time, constant connectivity, and information overload on students and educators, exploring strategies to mitigate these challenges.

VII. Opportunities for Lifelong Learning

The digital learning experience opens avenues for lifelong learning and continuous skill development. This subsection will explore how digital platforms have facilitated ongoing education, professional development, and the acquisition of new skills beyond traditional academic settings.

VIII. Impact on Academic Performance

The research will assess the impact of the digital learning experience on academic performance. This includes an analysis of the correlation between digital engagement, grades, and overall student success, providing insights into the effectiveness of digital learning methodologies.

3.3 Proposed Solution

Addressing the identified vulnerabilities and challenges within the digital learning landscape requires a strategic and robust approach. This section outlines a proposed solution designed to fortify data architectures, enhance cybersecurity measures, and ensure the integrity of grade distribution schemes within West African universities.

3.3.1 Comprehensive Security Model

In response to the identified vulnerabilities within West African universities' digital learning landscapes, a Comprehensive Security Model is proposed. This model aims to establish a robust and adaptive security infrastructure, safeguarding data architectures, and fortifying against cyber threats (Figure 1). The components of this model include:

Encryption Protocols: To ensure the confidentiality and integrity of data, the implementation of advanced encryption protocols is paramount. This involves encrypting data both in transit and at rest, utilizing industry-standard algorithms. The adoption of encryption mechanisms will secure sensitive information from unauthorized access or tampering.

Multi-Factor Authentication (MFA): Enhancing user authentication is critical to thwarting unauthorized access attempts. MFA, incorporating factors such as passwords, biometrics, or security tokens, adds an additional layer of protection. This reduces the risk of compromised user credentials and strengthens overall system security.

Role-Based Access Control (RBAC): offers a robust framework that brings about numerous advantages to security management within systems. Its structured approach simplifies administration by consolidating access control under predefined roles, mitigating the complexities of assigning individual permissions. This not only streamlines the process but also significantly reduces the likelihood of human error and unauthorized access. RBAC enhances security by precisely aligning permissions with job functions, fostering a principle of least privilege where users only gain access necessary for their roles, minimizing potential vulnerabilities. Its scalability and adaptability empower organizations to efficiently manage access rights, seamlessly accommodating changes in personnel or roles by adjusting role

assignments. RBAC stands as a cornerstone of access control, promoting a balance between stringent security measures and operational efficacy.

Regular Security Audits: A proactive approach is taken through regular and thorough security audits. These audits assess the effectiveness of existing security measures, identify potential vulnerabilities, and ensure compliance with cybersecurity best practices. Continuous monitoring and assessment contribute to a dynamic and resilient security posture.

Incident Response Plan:
Preparation for cybersecurity incidents is addressed by formulating a well-defined incident response plan. This plan outlines the steps to be taken in the event of a security breach, ensuring a swift and coordinated response to mitigate potential damages. Regular drills and updates refine the incident response strategy.

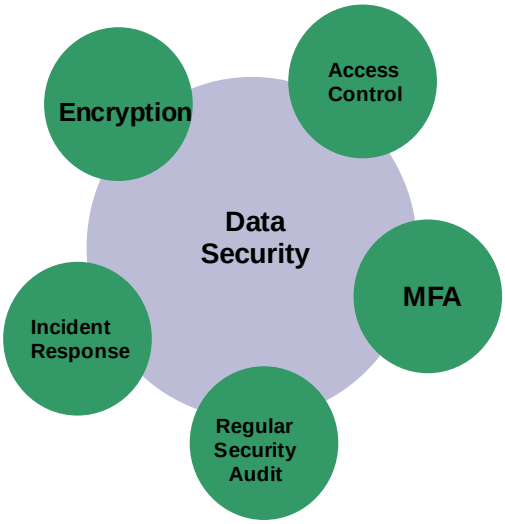


Figure 1: Comprehensive Security Model

3.3.2 Adaptive Data Architecture

In response to the diverse landscape of West African universities, an Adaptive Data Architecture is proposed to fortify the foundations of digital learning environments. This framework is designed to ensure scalability, resilience, and efficiency in managing academic data (Figure 2). The key components of the Adaptive Data Architecture include:

Scalable Infrastructure: Recognizing the fluctuating demands of digital learning, the architecture incorporates a scalable infrastructure. This entails the ability to dynamically adjust resources to accommodate variations in user activity, ensuring optimal performance during peak usage periods and efficient resource utilization during low-demand periods.

Cloud Integration: Strategic integration of cloud technologies forms a pivotal element of the Adaptive Data Architecture. Leveraging cloud services facilitates enhanced storage capacity, accessibility, and seamless data backup capabilities. This integration provides flexibility, scalability, and cost-effectiveness in managing academic data.

Redundancy Measures: To mitigate the impact of system failures or cyber-attacks, the architecture incorporates redundancy measures. Redundancy ensures that critical data is duplicated and distributed across multiple servers or locations, reducing the risk of data loss and enhancing overall system reliability.

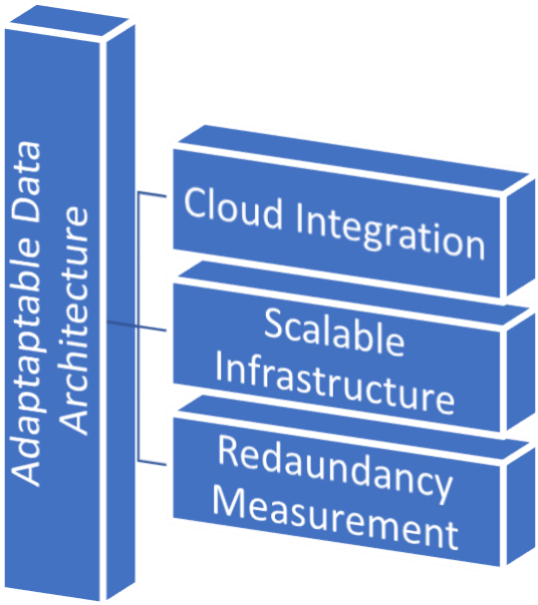


Figure 2: Adaptable Data Architecture

3.3.3 Secure Grade Distribution Scheme

In the pursuit of fortifying the grade distribution process within West African universities, a Secure Grade Distribution Scheme is proposed. This scheme integrates advanced cryptographic techniques, including AES (Advanced Encryption Standard), HSM (Hardware Security Module), Diffie-Hellman key exchange, and MIC (Message Integrity Check) verification. The amalgamation of these elements aims to ensure the confidentiality, integrity, and secure transmission of academic grades.

AES Encryption: The utilization of AES encryption stands as a foundational element in securing grade distribution. AES, a symmetric encryption algorithm, ensures the confidentiality of transmitted grades. Each grade is encrypted using a unique key, providing a robust defence against unauthorized access or tampering.

Hardware Security Module (HSM): To elevate the security posture, a Hardware Security Module is integrated into the scheme. HSM serves as a secure enclave for storing cryptographic keys, preventing unauthorized access. By utilizing HSM, the scheme enhances key management practices, safeguarding encryption keys from potential vulnerabilities.

Diffie-Hellman Key Exchange: The Diffie-Hellman key exchange protocol is incorporated to establish secure communication channels. This ensures that the encryption keys are exchanged securely between the sender and receiver without the risk of interception. Diffie-Hellman enhances the confidentiality of the grade distribution process.

Message Integrity Check (MIC) Verification: MIC verification is employed to ensure the integrity of transmitted grades. This involves attaching a cryptographic hash value to each grade, allowing the recipient to verify the authenticity of the received data. MIC verification acts as a crucial defence against any unauthorized alterations during transmission.

3.3.3.1 Benefits of the Secure Grade Distribution Scheme:

1. Confidentiality: AES encryption ensures that grades are transmitted confidentially and securely.

2. Key Protection: HSM safeguards cryptographic keys, reducing the risk of key compromise and unauthorized access.
3. Secure Communication Channels: Diffie-Hellman key exchange establishes secure channels for key transmission, enhancing overall communication security.
4. Data Integrity: MIC verification provides a robust mechanism for ensuring the integrity of transmitted grades, minimizing the risk of tampering.

3.3.3.2 Implementation Considerations:

1. Key Management: A robust key management strategy ensures the secure generation, distribution, and storage of encryption keys.
2. User Authentication: Implementing strong user authentication mechanisms ensures that only authorized individuals have access to grade distribution processes.

3.4 Tools used in the implementation.

In the implementation phase of this research, various tools were employed to facilitate the development and assessment of the proposed data-centric architecture and security model. The following tools played a crucial role in ensuring the effectiveness and reliability of the implemented solutions:

1. KConnect: KConnect served as the primary connector software in the data-centric architecture. Its robust capabilities and compatibility made it an ideal choice for seamlessly integrating diverse data sources and ensuring efficient communication between different components of the architecture.
2. Apache Ranger and Apache Atlas: These tools were instrumental in enforcing data governance and security measures within the architecture. Apache Ranger provided fine-grained access control and policy enforcement, while Apache Atlas facilitated metadata management and lineage tracking, enhancing overall data governance.
3. Hybrid Storage Solution: The implementation leveraged a hybrid storage approach, combining various storage solutions such as Network-Attached Storage (NAS), Storage Area Network (SAN), Cloud Storage, Local Storage, and Tape Storage. This

combination was essential for optimizing performance, ensuring cost efficiency, and enhancing data availability in the digital learning environment.

5. TensorFlow: TensorFlow, a powerful open-source machine learning library, was utilized for data analytics within the architecture. Its versatility and scalability allowed for the development of advanced analytics models, contributing to informed decision-making based on the analyzed educational data.
6. Kibana & Elastic Search: These tools were employed in the creation of user-centric applications. Kibana, with its visualization capabilities, and Elastic Search, providing efficient data retrieval, collectively enhanced the user experience by delivering intuitive and responsive applications.
7. Kafka: Kafka played a pivotal role in the implementation of the messaging system, ensuring seamless communication and data transfer between different components of the architecture. Its distributed and fault-tolerant nature contributed to the reliability of data streaming processes.
8. Data as a Service (DaaS): The concept of Data as a Service was integrated into the architecture, allowing for on-demand access to data. This facilitated efficient data sharing and collaboration among users, contributing to a more dynamic and interactive learning environment.

These tools collectively formed a cohesive and integrated technological ecosystem, enabling the development and deployment of a robust data-centric architecture tailored to the unique requirements of digital learning universities in West Africa. Their functionalities spanned from ensuring data security and governance to optimizing data storage, analytics, and user-centric applications, contributing to the overall success of the implemented model.

3.5 Approach and Technique(s) for the Proposed Solution

The successful implementation of the proposed Comprehensive Security Model, Adaptive Data Architecture, and Secure Grade Distribution Scheme within West African universities necessitates a systematic approach and the application of specific techniques. The chosen approach is structured and methodical, incorporating a series of well-defined steps and techniques tailored to each component of the overall solution.

3.5.1 Comprehensive Security Model

3.5.1.1 Approach:

1. Risk Assessment: Conduct a thorough risk assessment to identify potential threats and vulnerabilities specific to the digital learning environment of West African universities. Collaborate with cybersecurity experts and stakeholders to comprehensively analyze the risk landscape.
2. Baseline Security Measures: Establish baseline security measures by defining access controls, implementing network segmentation, and ensuring regular security updates across the digital infrastructure. Develop a foundational security framework to address common vulnerabilities.
3. Implementation of SIEM: Integrate a Security Information and Event Management (SIEM) system to centralize and analyze security event data in real-time. Configure SIEM to collect and correlate logs from various sources, allowing for proactive threat detection and response.
4. Endpoint Protection Deployment: Deploy advanced endpoint protection solutions on all devices within the digital learning environment. These solutions should encompass antivirus, anti-malware, and intrusion detection capabilities to safeguard against a spectrum of cybersecurity threats.

3.5.1.2 Techniques:

Penetration Testing: Conduct regular penetration testing exercises to simulate cyber-attacks and identify potential weaknesses in the security infrastructure. This technique provides insights into system vulnerabilities and ensures proactive security enhancements.

Continuous Monitoring: Implement continuous monitoring using SIEM tools to detect and respond to security incidents promptly. Real-time analysis of security events enables rapid incident response, reducing the impact of potential cyber threats.

Security Awareness Training: Provide security awareness training for faculty, staff, and students to promote a culture of cybersecurity. Educate users on recognizing phishing attempts, following secure practices, and reporting security incidents.

3.5.1.3 Benefits of the Approach and Techniques:

- Holistic Security Coverage: The approach ensures a holistic security coverage by addressing risks comprehensively and implementing baseline measures across the digital learning environment.
- Proactive Threat Mitigation: Techniques such as penetration testing and continuous monitoring contribute to proactive threat mitigation, allowing for the identification and remediation of security issues before they escalate.
- User Empowerment: Security awareness training empowers users to actively contribute to the security posture, fostering a collaborative approach to cybersecurity within the academic community.

3.5.2 Adaptive Data Architecture

3.5.2.1 Approach:

1. Data Classification: Initiate a comprehensive data classification process to categorize information based on sensitivity. This process will inform the implementation of security measures commensurate with the importance and confidentiality of the data.
2. Cloud Integration Strategy: Develop a strategic plan for seamless integration with reputable cloud service providers, such as Amazon Web Services (AWS) or Microsoft Azure. This strategy ensures scalable storage, efficient resource management, and reliable data backup capabilities.
3. Containerization Implementation: Implement containerization platforms like Docker or Kubernetes to enhance the deployment and scalability of applications within the adaptive data architecture. Containerization facilitates efficient resource utilization and accelerates the development lifecycle.
4. Load Balancing Configuration: Configure load balancing solutions, utilizing tools like HAProxy or Nginx, to optimize the distribution of network traffic. This ensures that the digital learning environment maintains optimal performance even during varying workloads.

3.5.2.2 Techniques:

- Data Encryption: Implement robust encryption mechanisms for sensitive data stored in the cloud. Encryption safeguards data confidentiality during transmission and storage, aligning with regulatory requirements and security best practices.
- Container Orchestration: Leverage container orchestration tools such as Kubernetes to automate the deployment, scaling, and management of containerized applications. This technique streamlines operations and enhances the adaptability of the adaptive data architecture.
- Regular Security Audits: Conduct regular security audits to assess the effectiveness of implemented security measures. These audits help identify potential vulnerabilities and ensure ongoing compliance with security standards.

3.5.2.3 Benefits of the Approach and Techniques:

- Scalability and Efficiency: The approach ensures the scalability of data storage and efficient resource management, allowing the digital learning environment to adapt to varying workloads.
- Resource Utilization: Containerization and load balancing techniques optimize resource utilization, enhancing the overall performance of applications and services.
- Adaptability to Changes: Techniques such as container orchestration enhance the adaptability of the architecture to changes in user demand and technological advancements.

3.5.3 Secure Grade Distribution Scheme

3.5.3.1 Approach:

1. Key Management Plan: Develop a robust key management plan to govern the generation, distribution, and secure storage of encryption keys within the Secure Grade Distribution Scheme. This plan ensures that cryptographic keys remain confidential and are appropriately managed throughout their lifecycle.
2. Integration of Cryptographic Libraries: Integrate advanced cryptographic libraries supporting the AES encryption algorithm, Hardware Security Modules (HSMs), and Diffie-Hellman key

exchange. These libraries form the foundational elements of the scheme, providing the necessary cryptographic functions for secure grade distribution.

3. Implementation of MIC Verification: Incorporate hashing libraries for Message Integrity Check (MIC) verification into the grade distribution process. This technique ensures the integrity of transmitted grades by attaching cryptographic hash values, allowing recipients to verify the authenticity of the received data.

3.5.3.2 Techniques:

- Diffie-Hellman Key Exchange: Implement the Diffie-Hellman key exchange technique to securely exchange encryption keys between the sender and receiver. This ensures a secure and confidential key distribution process within the grade distribution scheme.
- Hardware Security Module Usage: Utilize Hardware Security Modules (HSMs) for secure key storage and cryptographic operations. HSMs enhance the security of cryptographic keys by providing a dedicated and tamper-resistant hardware environment.
- AES Encryption: Implement AES encryption libraries to encrypt grades using symmetric key cryptography. This technique ensures the confidentiality of transmitted grades, preventing unauthorized access.

3.5.3.3 Benefits of the Approach and Techniques:

- Confidentiality and Integrity: The approach ensures the confidentiality and integrity of transmitted grades through the integration of cryptographic techniques such as AES encryption and MIC verification.
- Secure Key Exchange: Techniques like Diffie-Hellman key exchange facilitate a secure and confidential process for exchanging encryption keys, enhancing overall communication security.
- Key Protection and Management: The use of HSMs provides a secure enclave for key protection and management, minimizing the risk of key compromise and unauthorized access.

3.6 Research Design

The research design for this study draws inspiration from the framework proposed by Georgiadou et al. (2021), encompassing a systematic approach to ensure methodological rigour. The distinct phases, namely survey methodology, case studies for comparative analysis, and subsequent data analyses, are strategically aligned to address the research questions effectively.

3.6.1 Participants and Sampling

In alignment with the peculiarities of the survey, targeted participants included technical staff directly involved in managing learning management systems, university websites, and portals, as well as directors of IT units and academic planning. A deliberate sampling strategy aimed to secure a robust dataset, targeting a minimum of 1,000 responses from at least 90 institutions, constituting 70% of the 128 West African universities registered with the Association of African Universities (AAU, 2022).

3.7 Data Collection

3.7.1 Survey Methodology

The survey methodology served as the primary data collection technique, driven by a comprehensive set of 20 survey questions (SQs) designed in both English and French languages (Appendix A). Each question was meticulously crafted to address specific research questions (RQs), covering diverse aspects such as data architectures, cyber threats, countermeasure techniques, and the integration of data in decision-making.

3.7.1.1 Validity Testing

Before wide dissemination, rigorous validity testing involved a diverse group comprising survey specialists, experienced researchers, certified security and technology officers, and non-technical staff. This phase employed respondent debriefing, cognitive interviewing, think-aloud, and verbal probing techniques to refine the survey instrument. Feedback from this phase informed the development of the final survey version (see Appendix A).

3.7.1.2 Dissemination and Analysis

The survey, initiated on February 26th, 2022, was disseminated to West African universities, both public and private, through email and WhatsApp channels. The three-month circulation period from March 1st to May 31st, 2022, allowed for comprehensive data collection. Specific eligibility criteria limited participation to technical staff, emphasizing the importance of their roles in technological infrastructure.

A total of 1,164 responses were received from 93 universities, representing approximately 72% of West African universities registered with the AAU. To ensure data integrity, duplicate responses were avoided, and 109 responses indicating the absence of workshops or training during the COVID-19 pandemic were excluded. The remaining 1,055 responses formed the basis for the study.

Utilizing a four-point Likert scale for nuanced responses, where Extensively = 4, Moderately = 3, A little = 2, and Not at all = 1, facilitated a nuanced understanding of participant perspectives. The collected data underwent analysis using SPSS, ensuring a robust and systematic exploration of the research questions.

3.7.2 Methodology for Comparative Analysis

The comparative analysis methodology commenced with the identification of pertinent keywords, namely (i) e-learning or "online learning" or "digital learning," (ii) solutions, and (iii) "use case." These keywords were employed to formulate a comprehensive search string: (e-learning or "online learning" or "digital learning") and solutions and "use case." The Google search conducted on November 23, 2021, produced 38 entries, all of which were meticulously downloaded into Zotero, a reference management application.

Subsequently, on November 25, 2021, a meticulous screening process was initiated to ensure alignment with our research objectives. Out of the initial pool, 18 papers were excluded as they were deemed irrelevant to the research. Simultaneously, 20 publications were identified as eligible, contributing to a total of 109 use cases, each associated with a specific e-learning solution and meticulously mapped. This comprehensive analysis phase concluded on November 28, 2021.

Building upon this foundation, a thorough investigation into the data architecture of each mapped e-learning solution transpired on December 15, 2021. This investigation involved scrutinizing the websites and scrutinizing published white papers. Among the 20 identified e-learning solutions, 14 were found to employ data-driven architecture, while the remaining six utilized data-centric architecture. This phase of investigation was successfully concluded on December 25, 2021.

Advancing the research, a total of 983 user reviews were collected from the e-learning industry, with 696 emanating from identified data-driven e-learning solutions and 287 from data-centric counterparts. To analyse this voluminous dataset, a conceptual framework was developed, aligning with current e-learning requirements obtained from [5], [23]–[25] (Figure 3). The analysis transpired from December 26, 2021, to January 02, 2022.

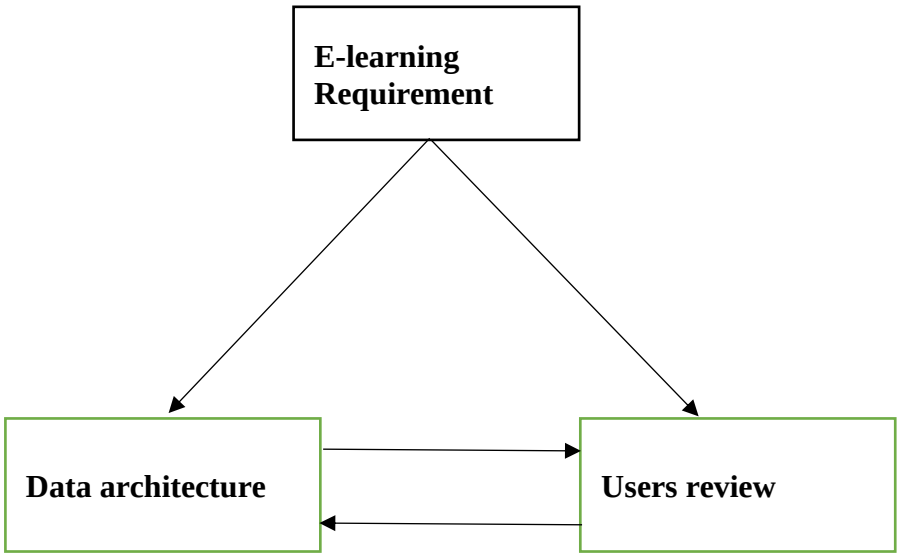


Figure 3. Proposed conceptual framework.

CHAPTER FOUR: RESULTS

4.1 Preamble

The fourth chapter of this thesis embarks on the practical implementation of the proposed frameworks, Comprehensive Security Model, Adaptive Data Architecture, and Secure Grade Distribution Scheme, within the dynamic context of West African universities' digital learning settings. This chapter serves as the bridge between theoretical concepts and real-world application, elucidating the strategic deployment strategies, the evaluation criteria, and the anticipated challenges inherent in implementing these innovative solutions.

As we delve into the practical realm, it is imperative to acknowledge the symbiotic relationship between theory and application. The envisaged security enhancements, data architecture adaptability, and secure grade distribution scheme are about to undergo a transformative journey from conceptualization to operationalization. This chapter provides a comprehensive account of the hands-on aspects of translating theoretical constructs into tangible solutions.

The successful implementation of these frameworks requires a meticulous approach, considering the unique nuances of each university's digital learning infrastructure. From configuring security measures to optimizing data architecture and securing grade distribution, this chapter navigates through the details of translating theoretical excellence into practical realities.

By detailing the step-by-step procedures, the selection and configuration of tools, and the nuances of integrating these solutions into existing university systems, this chapter aims to serve as a practical guide for information technology professionals, academic administrators, and other stakeholders involved in the implementation process. It is a testament to the commitment to fortifying the digital learning landscape and ensuring the integrity, security, and adaptability of academic processes.

In essence, this chapter signifies the convergence of vision and action, theory and practice, as we embark on the journey of transforming West African universities' digital learning environments into fortified bastions of academic excellence, resilience, and security.

4.2 Survey Findings

4.2.1 Overview of Data Architectures in West African Universities

Different researchers have established definitions for data architectures (DA), but the definition by (Zheng et al., 2010) is the one that is most frequently used. It describes DA as a collection of models, policies, guidelines, and standards that regulate the collected data types and how they are organized, integrated, stored, and utilized in data systems and organizations. According to (Ascend, 2020; Carol, 2021; Kampakis, 2018; Sinan, Degila, et al., 2022a), between 400 BC and 2022, DA progressed through four major stage:

1. Traditional architecture
2. Data-informed architecture
3. Data-driven architecture
4. Data-centric architecture

Our study will concentrate on data-informed, data-driven, and data-centric architectures because these are the only ones now in use by businesses and institutions (Sinan, Degila, et al., 2022a); the following are definitions taken from the literature:

- Data-informed Architecture: Data is collected from various sources, including flash drives, computers' internal and external hard drives, and so on. The data is analyzed using a spreadsheet, and the results are used as inputs in the decision-making (Ascend, 2020).
- Data-driven Architecture: In this approach, algorithms are utilized to generate decisions based on the data gathered from several data silos, including the cloud, data lakes, and other sources (Alfonso, 2018). (Kampakis, 2018) defines it as a DA in which storage devices or silos are scattered across several places and algorithms are used to preserve, analyze, and derive decisions from the analysis result. It is defined as a distributed storage architecture employing technology to gather and analyze data to make better business decisions (Kampakis, 2018).
- Data-centric Architecture: (Alfonso, 2018) refers to a system in which data is the primary and permanent asset, whereas applications come and go. In (Vista, 2021) and (Dave, 2020), organizations and institutions create a single data model that is shared by all of the organization’s information systems, data science is used as the bedrock for decision-making, and all data are linked and connected using a graph database to eliminate data silos and redundancy.

4.2.2 Demography

The first part of the survey is for demography including six survey questions, this aids in getting the descriptive data of the Universities and participants' behaviors towards securing their university data. Figure 1 presents the breakdown of the universities surveyed according to their countries with Nigeria being the highest with forty-nine (49) institutions, Ghana 23, Gambia 3, Senegal 3, Sierra leone 3, Burkina Faso 2, Cote d'voire 2, Niger 2, Togo 2. Benin, Liberia, Mauritania and Mali with the fewest amount of one (1) each, making a total of ninety-three Universities (Figure 4).

Additionally, seventy-seven (77) are public Universities and sixteen (16) are private, these institutions employed different modes of delivery (MOD), 44.4 % of the responses came from universities using face-to-face, 45.4% from e-learning institutions and 10.1% from blended MOD institutions, and all universities regularly create vast amounts of data as a result of the plethora of online activity. Of the respondents, 10.5% claimed their institutions only complete applications (A) online, compared to 56.6% who completed application and registration (AR), 6.6% who completed applications, registrations, and examinations (ARE), 8.9% who agreed that their institutions are always online for applications, registrations, and lectures (ARL), and 24.4% who agreed on applications, registrations, lectures, and examinations (ARLE) (Table 1).

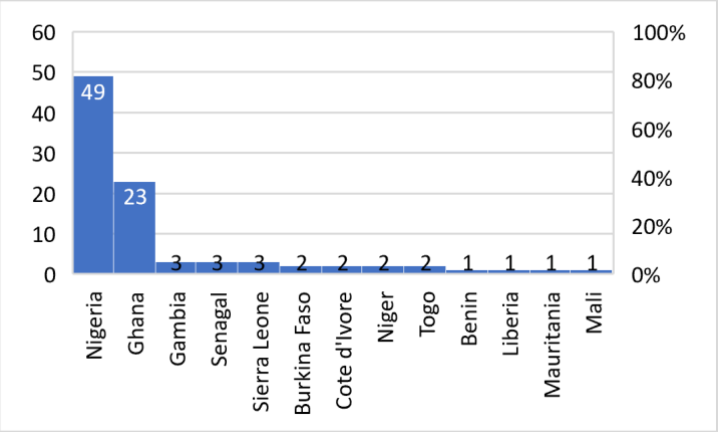


Figure 4 Number of universities according to countries

This demonstrates the significant reliance on online resources for the efficient operation of WAU. In terms of DA, 4.1% of the participants believed they employed data-centric architecture, 24.1% data-driven architecture and 71.8% data-informed architecture (Table 3)

Table 3 Demography of Universities

Online Activities			
Frequency	**Frequency**	**Percent**	**Cumulative percent**
Application	111	10.5	10.5
Application and Registration	591	56.0	66.5
Application, Registration and Examination	6	.6	67.1
Application, Registration and Lectures	94	8.9	76.0
Application, Registration, Lectures and Examination	253	24.0	100.0
Total	1055	100.0	
Mode of Delivery			
Blended learning	117	10.1	10.1
E-learning	518	44.5	54.6
Face-to-face learning	529	45.4	100.0
Total	1164	100.0	
Data Architecture			
Data-Centric Architecture	43	4.1	4.1
Data Driven Architecture	254	24.1	28.1
Data Informed Architecture	758	71.8	100.0
Total	1055	100.0	

The survey received huge responses from both males and females, 76% are males and 24% are females, this is of particular importance, as the ratio of females to males is the ideal ratio for productive work in a cybersecurity environment (Fatokun et al., 2019). Additionally,18.6% of participants were under 25 years, followed by 39.6% between 26 and 35 years, 27.7% from 35 to 45 years, 10% from 46 to 55 years, and 4.7% from participants over 56years of age, Figure 5 shows a histogram of age having mean of 2.43, this is vital it entails that, universities staff have the ideal age to learn new emerging cybersecurity techniques.

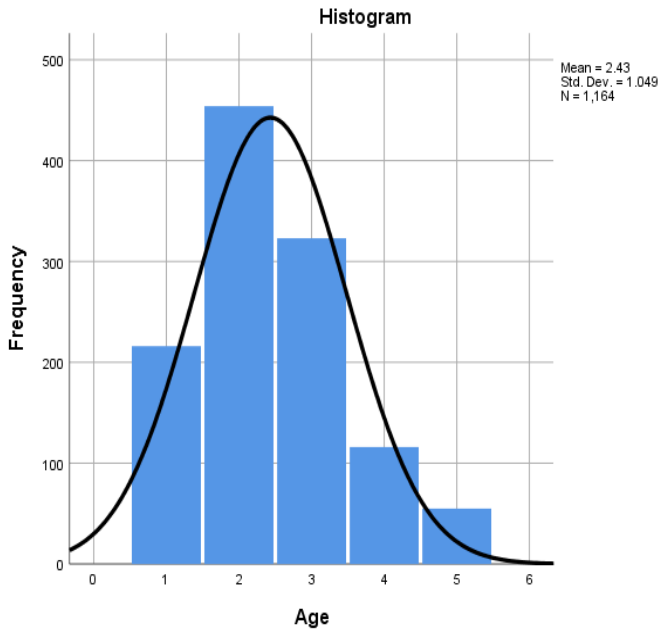


Figure 5. Histogram of age analysis

Furthermore, among the participants, 19.8% have a diploma, 46.1% have a bachelor's degree, 20.6 % have completed their master's, and 13.3% have a PhD (Table 4)

4.2.3 Data Application and Usability

In this survey, participants were given several questions on the use of data and analysis results in making decisions using a four-point Likert scale. The responders were initially questioned on the types of data they gather for analysis prior to making decisions, the tools they used to execute the analysis, and the type of decisions they made.

Table 4 Demography of participants

Frequency	Frequency	Percent	Cumulative percent
Gender			
Female	279	24.0	24.0
Male	885	76.0	100.0
Total	1164	100.0	
Age			
26 – 35 years	454	39.0	39.0
36 – 45 years	323	27.7	66.8
46 – 55 years	116	10.0	76.7
56 and above	55	4.7	81.4
Below 25 years	216	18.6	100.0
Total	1164	100.0	
Qualification			
Bachelor	537	46.1	46.1
Diploma	231	19.8	66.0
Masters	240	20.6	86.6
PGD	1	.1	86.7
PhD	155	13.3	100.0
Total	1164	100.0	

Figure 6 presents the details of the type of data used for analysis for WAU, 35.1% believed their institutions don't use data at all for decision-making they rely on their gut feeling and experience, 36.1% claimed they use data about what happened in the recent past (e.g last year or last quarter), 21.1% agreed that their Universities use past and recent data including some longer-term trends analysis and 7.6% said they use past, present and forward-looking data.

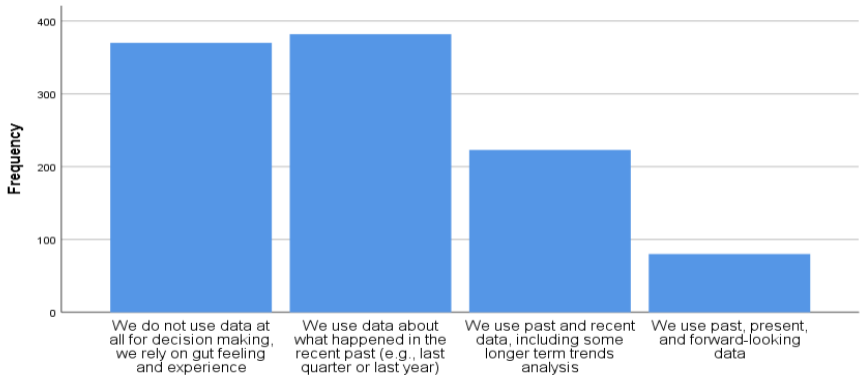


Figure 6 Breakdown of the types of data employed by WAU

Table 5 shows descriptive statistics of the tools employed for analysis by WAU, spreadsheets (e.g charts, counts, tables) have a mean of 3.09, website analytics (e.g google analytics) with 2.22, database (e.g CRM analytics, reports) with 2.65 and specialised tools (e.g SAS, R, Stata, Python, SPSS, GIS mapping) has 2.57.

Table 5 Descriptive statistics of tools used for analysis

Tools	**N**	**Minimum**	**Maximum**	**Mean**	**Std. Deviation**
Spreadsheet	1055	1	4	3.09	.951
Website Analytics	1055	1	4	2.22	1.208
Database	1055	1	4	2.65	.933
SpecializeTools	1055	1	4	2.57	.974
Valid N (listwise)	1055				

The data analysis result is used by WAU to make decisions on different categories, in terms of academic development decisions it has a mean of 2.74, employment 2.70, environmental impacts 2.70, other societal impacts 2.68, research opportunities 2.70 and student satisfaction has 3.11 (Table 6)

Table 6. Descriptive statistics on the use of data analysis results

Data Usage	**N**	**Mean**	**Std. Deviation**
Academic Development	1055	2.74	.913
Employment	1055	2.70	.971
Environmental Impacts	1055	2.70	.964
Other Societal Impacts	1055	2.68	.956
Research Opportunities	1055	2.70	.975
Students Satisfaction	1055	3.11	1.022
Valid N (listwise)	1055		

4.2.4 Cyberattacks and Countermeasures.

During the Covid-19 pandemic, WAU were severely targeted by cyberattacks. 87.4% of the responders indicated that they were victims of cyberattacks, and 12.6% were not. Of the victims who are knowledgeable enough about the security vulnerabilities at their institutions, 621 agreed their institutions were attacked by SQL injection, 752 by a denial-of-service attack, 565 by ransomware, 451 by a virus, 214 by a worm, 335 by a phishing attack, and 1 participant reported not knowing about any cyberattacks (Figure 7).

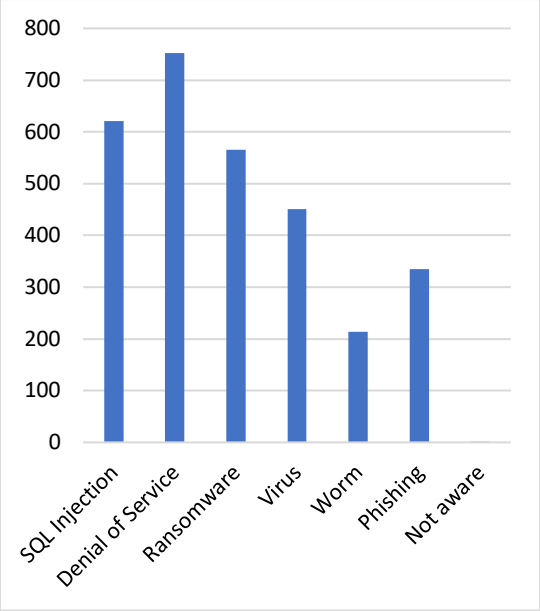


Figure 7. Summary of cyberattacks faced by WAU

Moreover, the participants receive cybersecurity training, but only 8.1% complete it after 3 months, 10.2% do so after 6 months, 40.3% do so after 12 months, and 41.1% have never attended any cybersecurity training (Table 7)

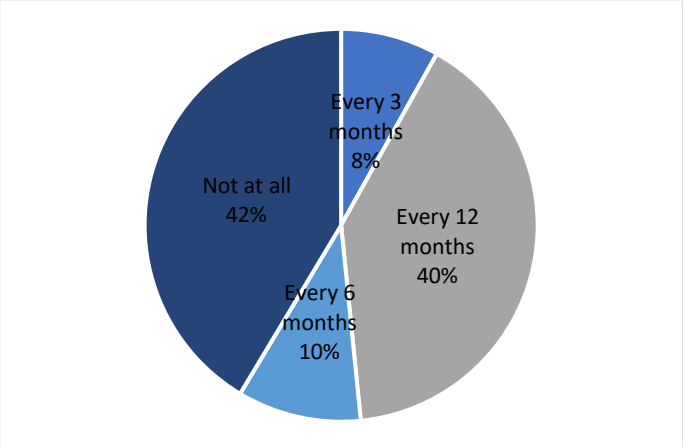


Figure 8. Summary of staff cybersecurity training

Moreover, a variety of countermeasures are used by WAU, Table 6 breaks down these techniques. These institutions used a variety of techniques to ensure secure cyberspace for learning, and many participants (52.9%) claimed their institutions only used firewalls and antivirus software for security, while 0.1% thought their institutions used firewalls, intrusion detection systems, and intrusion prevention system.

Table 7 Countermeasures

Countermeasures Technique	Frequency	Per cent	Cumulative Percent
Anti-virus	28	2.7	2.7
Anti-virus, Intrusion detection system	173	16.4	19.1
Anti-virus, Intrusion detection system, Intrusion prevention system	12	1.1	20.2
Anti-virus, Intrusion prevention system	4	.4	20.6
Firewall	22	2.1	22.7
Firewall, Anti-virus	558	52.9	75.5
Firewall, Anti-virus, Intrusion detection system	176	16.7	92.2
Firewall, Anti-virus, Intrusion detection system, Intrusion prevention system	38	3.6	95.8
Firewall, Anti-virus, Intrusion prevention system	9	.9	96.7
Firewall, Intrusion detection system	13	1.2	97.9
Firewall, Intrusion detection system, Intrusion prevention system	1	.1	98.0
Intrusion detection system	9	.9	98.9
Intrusion detection system, Intrusion prevention system	8	.8	99.6
Intrusion prevention system	2	.2	99.8
Not known	2	.2	100.0
Total	1055	100.0	

Responders were asked about the level of satisfaction they had with their institution's data protection techniques; Table 8 shows that it has a mean 2.24.

Table 8 Descriptive Statistics on Satisfaction

Items	N	Mean	Std. Deviation
Satisfaction	1055	2.24	.795
Valid N (listwise)	1055		

4.2.5 Discussion

This study created a survey and distributed it to WAU to determine the security vulnerability of their data architectures, techniques for preventing cyberattacks, and the effect of data analysis on decision-making. In this section, first and foremost, we will discuss demographic analysis, data analysis and usability, and cyberattacks and countermeasures.

Looking at figure 2's age analysis, it has a mean of 2.43 (std. Dev 1.049) which indicates that the majority age of the participants is 25-35 years, and 46.1% have bachelor's degrees, which is the perfect age and educational background for the staff to learn new skills for fending off cyberattacks. Furthermore, analysis demonstrates that the gender ratio is favorable for staff to co-exist for effective work in a cybersecurity environment (Fatokun et al., 2019). In addition, WAU has quickly made the switch to digital learning; 45.1% of the institutions surveyed used e-learning as a MOD, and every institution had at least one online activity. This makes it a challenge for both researchers and industries to provide safe and secure data architecture in this region.

In addition, WAU are always looking for research gaps that may be addressed by academic researchers in addition to staff employment, enrolling more students, and developing staff capacity. However, the results of this study indicate that, in order to run these Universities efficiently, there is a need to optimize the utilization of data analysis results. Moreover, Table 5 shows that data analysis results for academic development have a mean of 2.74, employment has a mean of 2.70, environmental impacts have a mean of 2.68, research opportunities have a mean of 2.70, and student satisfaction has a mean of 3.11; this demonstrates the specific areas that need improvement, particularly areas with less than 3.0. Additionally, the type of data acquired for analysis before decision-making and the tools used for analysis are also causes for concern. According to the study's findings, only 7.6% of participants believed their institutions used past, present, and future-looking data for analysis, while 35.1% agreed that they used their intuition and experience instead. Furthermore, with a mean of 3.09, the majority of participants chose to use spreadsheet software for data analysis, compared to less than 2.6 for the other tools, which is worrying. This creates a vacuum for WAU to enhance the type of data and analysis tools.

Findings show that WAU are always conducting activities online, be it application, registration, lectures or facilitation, or examination, which yields data generation and are yet to get a secured means of storing their data. Only 12.6% indicated that their universities were not victims of cyberattacks. These attacks are due to several factors particularly:

- Inability to upgrade their data architecture to the newest, this study finds out that 71.8% use DIA which is the most obsolete DA in existence, followed by DD with 24.1%, and

4.1% employed DCA which is the advanced DA in existence now and it is highly secured with few security vulnerabilities (Kim, 2019)

- The technical staff maintaining learning management systems, websites, and portals lack cybersecurity knowledge and training. This study reveals that 41.1% of the staff have never taken cybersecurity training, 40.3% have done so after every 12 months, 10.2% have taken it after 6 months, and 8.1% have taken it after 3 months. The training has a mean of 1.85, indicating that the majority of participants have never taken cybersecurity training (Table 6), and the analysis of the cybersecurity skills of the participants reveals they have a mean of 3.43, demonstrating the need for frequent training and workshops.
- Lack of adequate countermeasures to efficiently prevent and detect cyberattacks. The finding of this study shows that universities use several techniques when repelling cyberattacks, 52.9% use firewalls and anti-virus software which is not efficient, while 0.1% believed their institution employed firewalls, intrusion detection systems and intrusion prevention systems.

Additionally, on a scale of yes, neutral, and no, the participants were also asked to rate their level of satisfaction with their institution's countermeasures strategy. Analysis reveals that it has a mean of 2.24 (Table 6), indicating that the majority of participants are not satisfied with their institution's countermeasures strategy.

4.3 Comparative Analysis

4.3.1 Introduction

This section serves as a comprehensive comparative analysis aimed at discerning the optimal data architecture for digital learning universities, specifically tailored to accommodate the multifaceted requirements of e-learning. To facilitate this evaluation, the study initiates the process by identifying and scrutinizing 109 distinct e-learning solution use cases. Through meticulous classification, each use case is systematically categorized based on the underlying data architectures employed.

To enrich the depth of this comparison, the study delves further into the practical insights garnered from the e-learning industry by procuring and analyzing 983 user reviews. This qualitative approach ensures a nuanced understanding of the user experience, shedding

light on the nuances and intricacies of various data architectures within the e-learning domain.

The identification and classification of e-learning solution use cases lay a robust foundation for the subsequent analysis. By dissecting each use case based on the employed data architecture, the study unveils patterns, strengths, and potential limitations associated with different approaches. This intricate examination is instrumental in formulating a nuanced understanding of the diverse landscape of data architectures in the context of digital learning universities.

The inclusion of 983 user reviews amplifies the comparative assessment, providing a qualitative dimension to the evaluation process. These reviews, sourced from within the e-learning industry, encapsulate real-world experiences and perspectives, offering valuable insights into the practical implications of various data architectures. Users' feedback becomes a crucial lens through which the study gauges aspects like user satisfaction, system performance, and overall efficacy, adding a layer of authenticity to the comparative analysis.

In essence, this section not only outlines a methodology for identifying and classifying e-learning solution use cases based on data architectures but also extends its reach into the realm of user experiences. By combining quantitative data on use cases with qualitative feedback from industry users, the study aspires to present a well-rounded and informed perspective on the most suitable data architecture for digital learning universities engaged in the dynamic landscape of e-learning.

4.3.2 E-learning Solution Use cases

Figure 9 presents a detailed breakdown of e-learning use cases, with employee training emerging most frequently at fourteen (14) occurrences. Following closely are customer training at thirteen (13), compliance training, and academic learning, both registering twelve (12) each. Employee onboarding and training companies are tied at eleven (11) each, while continuing education follows with seven (7) instances. Further down the list, extended enterprise and dealer training both have six (6), and channel training is documented at four (4). Immersive

learning and competency management share a count of four (4), while workforce development concludes the breakdown with three (3) instances.

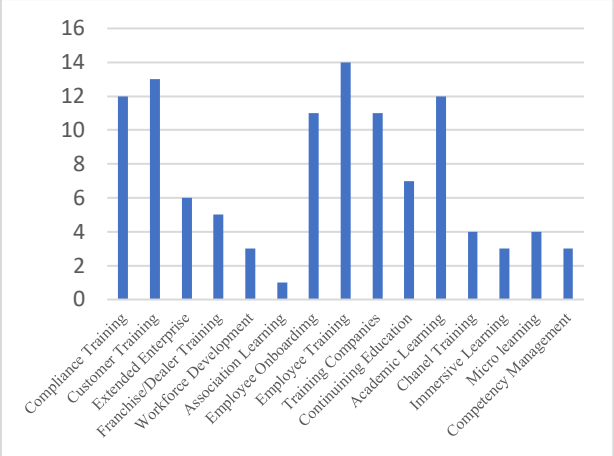


Figure 9. Use cases.

The use cases were then mapped to twenty (20) e-learning solutions (Table 2), with five (5) different solutions containing seven (7) use cases each, one (1) solution containing six (6) use cases, twelve (12) solutions containing five (5) use cases each, and two (2) solutions containing four (4) use cases.

Table 9. Use Cases

Reference(s)	E-learning solutions	Number of Use cases	Use Cases																
			Academic Learning	Association Learning	Compliance Training	Customer Training	Continuing Education	Chanel Training	Competency Management	Extended Enterprise	Employee Onboarding	Employee Training	Franchise Training	Immersive Learning	Microlearning	Training Companies	Workforce Development
(Ghatak, 2021)	Adobe Captivate Prime	5			X	X				X			X				X
(Lynch, 2021)	Absorb LMS	7	X		X	X				X	X	X				X	
(Kapadia, 2021)	GyrusAim	5			X	X						X					X
(Docebo, 2021)	Docebo	4			X	X					X	X					
(Gray, 2021)	Xperiencify	7	X				X	X			X			X	X	X	
(Jennifer, 2021)	Inquisiq	5	X		X	X						X				X	
(Brown, 2021)	Coassemble	7			X	X					X	X	X		X	X	
(Butler, 2021)	Nimble LMS	7			X	X					X	X				X	X
(Malekos, 2021)	LearnWorlds	5				X	X				X	X				X	
(Ponomarev, 2021)	Gurucan	4					X				X					X	X
(Media, 2021)	Eurekos LMS	5				X		X		X		X	X			X	
(Doust, 2021)	glo™ learn	7	X		X					X	X	X	X			X	
(Papagelis, 2021)	TalerntLMS	5			X	X				X	X	X					
(Bellaj, 2021)	Etakwin	5	X			X	X					X				X	
(Scott, 2021)	Thinkific	5	X			X	X						X			, X	
(Shodeinde, 2021)	Claned	5	X		X		X					X				X	
(Pappas, 2021)	Edysby	5	X	X	X		X		X								
(Ispring, 2021)	Ispring	5			X	X				X	X	X					
(learn upon, 2021)	LearnUpon LMS	6			X	X				X	X	X				X	
(Gogos, 2021)	Looop	5			X	X				X	X	X					

4.3.3 E-learning Solutions Mapping with Data Architectures

Table 10 shows the mapping of the e-learning solutions with data architectures, data-driven architecture accounting 70% of the mapping, including adobe captivate prime, absorb LMS, inquisiq, gurucan, eurekos LMS, glo™ learn, talerntLMS, etakwin, thinkific, claned, looop, canopyLAB, ispring, and learnUpon LMS, and data-centric architecture accounting for 30% of the mapping including gyrusAim, docebo, xperiencify, coassemble, nimble LMS, and learnWorlds.

Table 10 Data Architectures Mapping

Data architecture(s)	E-learning solutions
Data-driven architecture	Adobe Captivate Prime(Adobe, 2021; Ghatak, 2021) Absorb LMS (Inc, 2021; Lynch, 2021) Inquisiq (Jennifer, 2021; "Privacy Policy," 2021) Gurucan (Campus, 2021) Eurekos LMS (Bill, 2021) glo™ learn (GDPR, 2018; Policy, 2020) TalerntLMS (Papagelis, 2021; talent LMS, 2021) Etakwin (Bellaj, 2021) Thinkific (Scott, 2021; thinkific, 2020) Claned (claned, 2021; Shodeinde, 2021) Looop(Gogos, 2021) CanopyLAB (Hjorth, 2021) Ispring (Ispring, 2021) LearnUpon LMS (learn upon, 2021)
Data-centric architecture	GyrusAim (gyrus, 2021; Kapadia, 2021) Docebo (Docebo, 2021; docebo, 2021) Xperiencify (Gray, 2021; xper, 2020) Coassemble (Brown, 2021; pseudonymisation, 2020) Nimble LMS (Policy, 2018) LearnWorlds (CIO, 2021)

4.3.4 Comparative Analysis

In this sub-section, we present the comparison between data-driven and data-centric architectures. The data obtained for this study is presented in Table 4. Data-driven architecture received the most reviews, with 696 from 14 different e-learning solutions, while data-centric architecture received 287 from 6 other e-learning solutions.

Table 11. Users Review Data

Data architecture	Reference	Reviews	Total
Data-driven architecture	(Bellaj, 2021; claned, 2021; Doust, 2021; Ghatak, 2021; Gogos, 2021; Hjorth, 2021; Ispring, 2021; Jennifer, 2021; learn upon, 2021; Lynch, 2021; Media, 2021; Papagelis, 2021; Ponomarev, 2021; Scott, 2021)	Adobe Captivate Prime 63, Absorb LMS 20, Inquisiq 20, Gurucan 121, Eurekos LMS 16, glo™ learn 20, TalerntLMS 220, Etakwin 7, Thinkific 12, Claned 16, Looop 89, CanopyLAB 26, Ispring 29,LearnUpon LMS 37	696
Data-centric architecture	(Brown, 2021; Butler, 2021; Docebo, 2021; Gray, 2021; Kapadia, 2021; Malekos, 2021)	GyrusAim 64, Docebo 33, Xperiencify 77, Coassemble 36, Nimble LMS 40, LearnWorlds 37	287

Table 11 shows the comparison details based on e-learning requirements under data-driven architecture with 696 reviews. 69.7% of reviewers claimed it has real-time collaboration, 55.6

% claimed it gives students the ability to perform tasks anywhere, 39.7% claimed it is friendly with third-party applications, 45.8% suggested it has data integration, and 31.5% claimed it has a flexible environment. On the other hand, it has no data security, no data ownership, no data access permission, no data traceability, and data insight, according to 3.3%,10.9%, 41.5%, 2.2%, and 3.6%, respectively. Data-centric architecture received the fewest reviews (287), with 65.2 %, 37.9%, 41.4 %, 74.6 %, 93.3 %, 39.7%, 24 %, 27.2 %, 5.9%, and 5.2 % believing it has real-time collaboration, the ability for students to perform tasks anywhere, interoperability with other apps, data integration, customization, data ownership, data access permission, and data insight respectively. In comparison, 3.1% claimed it has no data security.

Table 12. Comparative Analysis

Data architecture(s)	**Data-driven architecture**			**Data-centric architecture**		
Number of reviews	696			287		
Analysis	Evaluation	Number of reviews	Percentage of the reviews	Evaluation	Number of reviews	Percentage of the review
Real-time collaboration	✓	485	69.7%	✓	187	65.2%
Ability for students to perform labs task anywhere	✓	387	55.6%	✓	109	37.9%
Interoperability with other apps	✓	276	39.7%	✓	119	41.4%
Data integration support	✓	319	45.8%	✓	214	74.6%
Flexible environment	✓	219	31.5%	✓	268	93.3%
Data security	X	23	3.3%	X	09	3.1%
Customization	✓	145	20.8%	✓	114	39.7%
Data ownership	X	76	10.9%	✓	69	24%
Data access permission	X	289	41.5%	✓	78	27.2%
Data traceability	X	15	2.2%	✓	17	5.9%
Data insight	X	25	3.6%	✓	15	5.2%

4.3.5 Discussion

This study undertakes a comprehensive examination, identifying and categorizing 109 e-learning solution use cases based on their respective data architectures. Additionally, a rich dataset of 983 user reviews from the e-learning industry was amassed, contributing valuable insights into the perceived efficacy of the identified data architectures. Subsequently, a meticulously crafted conceptual framework was developed, harnessing the wealth of user reviews to facilitate a nuanced comparison of the identified data architectures.

The framework, deployed to scrutinize both data-driven and data-centric architectures, unearthed noteworthy observations. Data-driven architecture, while exhibiting strengths, revealed limitations in critical aspects such as data security, data ownership, data access

control, data traceability, and data insight. Conversely, data-centric architecture displayed a limitation primarily in the domain of data security.

This discerning analysis positions data-centric architecture as the more fitting choice for e-learning environments. The identified strengths and limitations underscore the importance of tailoring data architecture to the unique demands and sensitivities of the e-learning landscape. Considering the findings, this study advocates for a paradigm shift towards embracing data-centric architecture to optimize the overarching efficacy and security of e-learning solutions.

4.4 Data-Centric Model

Data-centric architecture is an approach to data management that places data at the center of decision-making processes (Sinan, Degila, et al., 2022b). In education, this approach involves the creation of a data-driven culture where data is treated as an asset and is leveraged to drive decision-making. The key principles of data-centric architecture in education include data governance, data quality, data integration, and data analytics (Figure 10).

Data governance involves establishing policies and procedures for data management, ensuring compliance with privacy and security regulations, and promoting data sharing across different departments (Al-Naser et al., 2013). Data quality involves ensuring that data is accurate, complete, and consistent. Data integration involves consolidating data from different sources to create a single view of student data. Data analytics involves using advanced analytics to gain insights into student behavior and improve learning outcomes.

The benefits of data-centric architecture in education are numerous. It enables personalized learning, where learning resources are tailored to the needs of individual students. It also enables early identification of at-risk students, allowing for timely interventions to be made. Furthermore, it enables the optimization of resource allocation, where resources are allocated based on student needs and performance.

Data-centric architecture also promotes collaboration among stakeholders in education. For example, teachers can collaborate with other teachers to share best practices and improve

student outcomes. School administrators can collaborate with teachers to identify areas where resources need to be allocated to improve student outcomes. Additionally, data-centric architecture enables data sharing across different departments, which facilitates cross-functional decision-making.

However, there are also challenges associated with data-centric architecture in education. One challenge is ensuring data privacy and security. Another challenge is the complexity of data integration, where data is collected from different sources and in different formats. Additionally, there may be resistance to change from stakeholders who are accustomed to traditional data architectures.

Figure 10. Data-centric Architecture (Arora et al., 2018)

4.4.1 Components of data-centric architecture

i. Data Sources: In the foundation of a data-centric architecture lie the data sources, diverse origins of information ranging from databases and applications to cloud services and IoT devices. These sources are instrumental in providing a comprehensive view of data, encompassing structured and unstructured formats, batch or real-time streams, and varying data quality considerations.

ii. Connectors: Acting as crucial bridges between disparate data sources and the central data hub, connectors play a vital role in facilitating the extraction, transformation, and loading processes. They ensure a smooth and seamless flow of data from source systems to the architecture, managing complexities related to data extraction, format transformation, and secure transfer.

iii. Data Hub: At the heart of the architecture lies the data hub, serving as the central repository where data from diverse sources converges. It acts as a unified storage and processing hub, providing efficient data management, organization, and accessibility. A well-designed data hub supports scalability, ensuring the architecture's ability to handle growing volumes of data.

iv. Data Governance and Security: Data governance establishes policies, procedures, and controls for managing data throughout its lifecycle, ensuring compliance with regulations. Security mechanisms guarantee the confidentiality, integrity, and availability of data. These components are essential for maintaining trust in the accuracy and privacy of stored information.

v. Data as a Service: Data as a Service (DaaS) simplifies data consumption by providing on-demand access to specific data functionalities or datasets. It abstracts the complexities of data management, allowing users and applications to consume data without in-depth knowledge of underlying storage and processing intricacies. DaaS contributes to the agility of the architecture, focusing on data consumption rather than management.

vi. User-Centric Apps: In a data-centric architecture, user-centric apps are applications designed with a primary focus on user experience. Leveraging the available data, these apps provide valuable insights and facilitate decision-making. They include dashboards, reporting tools, or specialized interfaces tailored to user needs, enhancing the usability and utility of the architecture for stakeholders.

vii. Analytic Factory: The analytic factory represents a structured environment for developing, deploying, and managing analytics within the architecture. It streamlines the analytics process, from data preparation to model deployment, offering tools and processes for analytics lifecycle management. The goal is to create a systematic approach to generating actionable insights from data.

viii. Analytics as a Service: Extending the concept of DaaS to analytics functionalities, Analytics as a Service (AaaS) provides on-demand access to analytics tools, algorithms, or platforms. This allows users to perform complex analyses without managing underlying infrastructure. AaaS enhances the scalability and accessibility of analytics within the architecture, promoting flexibility and cost-effectiveness.

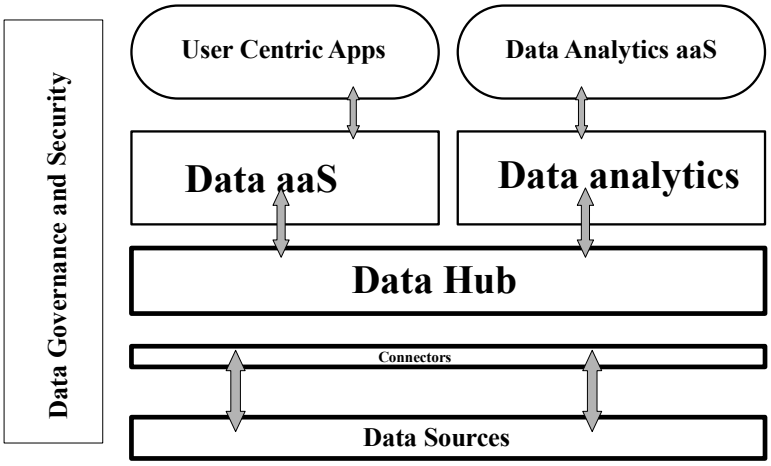


Figure 11 Components of Data-centric Model

Each of these components contributes to the cohesive and effective functioning of the data-centric architecture, ensuring it meets the demands of modern applications, particularly in contexts like digital learning universities (Figure 11).

4.4.2 Designing Data-centric Architecture Model

In the formulation of a specialized model intended for digital learning universities to effectively address the distinctive requirements prevalent in West African academic institutions, the incorporation of the following key components played a pivotal role. These components were strategically chosen to ensure a comprehensive and tailored approach to cater to the specific needs and challenges encountered within the realm of digital learning in the West African context (Figure 12).

i. Data Sources: In the intricate landscape of a digital learning university, a plethora of data sources collectively shape the foundation of a comprehensive data-centric architecture. At its core lies the Learning Management System (LMS), capturing user engagement metrics and course interactions, pivotal for assessing educational content effectiveness. Mobile applications tailored for education generate insights into user preferences and engagement patterns across diverse devices. The integration of Internet of Things (IoT) devices, administrative systems, and academic platforms contributes to a holistic understanding of the institution's operations. Social media engagement, library systems, and online collaboration tools provide additional layers of valuable data, reflecting broader trends, sentiment, and collaborative learning dynamics.

Research platforms, e-learning content platforms, and student information systems offer specialized data crucial for strategic decisions, innovation, and personalized learning approaches. Alumni engagement platforms, assessment tools, and testing platforms enrich the data landscape, providing insights into alumni contributions, academic proficiency, and continuous improvement areas.

ii. Connectors: In the realm of digital learning universities, the pivotal role of connectors within a data-centric architecture cannot be overstated. KConnect emerges as an exemplary connector, adept at seamlessly integrating diverse data sources prevalent in educational ecosystems. Its adaptability extends to Learning Management Systems, educational apps, IoT devices, administrative systems, and other academic platforms, fostering a cohesive data flow. Noteworthy for its flexibility in handling various data formats and robust security measures, KConnect ensures the synchronization of real-time data, contributing to streamlined decision-making and enhanced insights. As a linchpin in the architecture, KConnect epitomizes efficiency, reliability, and adaptability, addressing the dynamic needs of educational institutions in the digital age.

iii. Data hub: Within the data-centric architecture of digital learning universities, the Data Hub stands as a central nexus for managing and orchestrating data flow. In this context, the implementation of a hybrid storage approach is integral to the Data Hub's functionality. The hybrid storage system seamlessly integrates various storage solutions, including Network-Attached Storage (NAS), Storage Area Network (SAN), Cloud Storage, Direct-Attached Storage (DAS), and Tape Storage. This comprehensive strategy ensures optimal performance, scalability, and cost efficiency by balancing the strengths of each storage solution. Notably, all data stored in the local storage component of the hybrid model is fortified with AES encryption, bolstering data security and confidentiality. This encryption protocol serves as a robust safeguard, mitigating the risk of unauthorized access and enhancing the overall resilience of the digital learning university's data ecosystem. The Data Hub, fortified by hybrid storage and AES encryption, emerges as a strategic linchpin, fostering seamless data integration, accessibility, and security within the educational landscape.

iv. Kafka: In the data-centric architecture of digital learning universities, Kafka assumes a critical role as a distributed event streaming platform, meticulously designed by the Apache Software Foundation. Operating on a publish-subscribe model, Kafka becomes an indispensable component, orchestrating real-time data streams and fostering seamless communication across diverse applications and data sources within the

university ecosystem. Serving as the backbone of a robust and scalable data pipeline, Kafka excels in handling large volumes of streaming data, ensuring the instantaneous ingestion, processing, and dissemination of information—a pivotal capability in the dynamic realm of digital education. Beyond its prowess in real-time data management, Kafka acts as a reliable connector, facilitating the integration of varied data sources like Learning Management Systems, educational apps, IoT devices, administrative databases, and academic repositories. Kafka's fault-tolerant and distributed architecture guarantees data integrity and availability, positioning it as a linchpin in constructing resilient, responsive, and data-driven educational infrastructures for digital learning universities.

v. Data governance and Security: In the intricate landscape of data-centric architecture for digital learning universities, robust data governance and security measures are imperative. Apache Ranger and Apache Atlas emerge as stalwart guardians, orchestrating a formidable shield against unauthorized access, ensuring compliance, and fostering comprehensive data management. Apache Ranger stands as a sentinel, providing fine-grained access control and centralized security policies, allowing institutions to define, enforce, and audit data access policies seamlessly. Its capabilities extend to safeguarding sensitive information, mitigating risks, and upholding regulatory compliance within the educational data ecosystem. Complementing this protective bastion, Apache Atlas takes the reins in metadata management and lineage tracking. It meticulously catalogs and classifies data entities, offering a holistic view of the data landscape. This not only enhances transparency but also fortifies the ability to trace the origins and transformations of data—an invaluable asset in maintaining data quality and integrity. Together, Apache Ranger and Apache Atlas synergize to establish an unyielding fortress, instilling confidence in the secure handling of diverse data sources, from Learning Management Systems to administrative databases, ensuring the privacy and integrity of the wealth of information within the digital learning university domain.

vi. Data as a service: In the context of data-centric architecture, Data as a Service (DaaS) emerges as a pivotal component that revolutionizes how digital learning universities handle and deliver data. DaaS acts as a cloud-based service providing on-demand access to a variety of data sources, fostering seamless integration and utilization of data across different applications and systems. In the realm of digital learning, data sources such as Learning Management Systems (LMS), educational apps, Internet of Things

(IoT) devices, administrative records, and academic databases serve as examples of the rich and diverse data offerings that can be encapsulated by DaaS. The implementation of DaaS in the data-centric architecture ensures that users within digital learning universities can easily access, retrieve, and leverage data without the constraints of physical or geographical boundaries. This streamlined accessibility not only enhances the overall efficiency of educational processes but also promotes a more personalized and adaptive learning experience for students and educators alike. Furthermore, by encapsulating data from various sources, including hybrid storage solutions, DaaS contributes to the overarching goal of creating a unified and comprehensive data ecosystem within the educational landscape. In terms of security, DaaS platforms often integrate robust access controls, encryption mechanisms, and data governance frameworks, ensuring the confidentiality and integrity of sensitive information. As digital learning continues to evolve, the incorporation of Data as a Service stands as a pivotal step towards establishing a dynamic, interconnected, and secure data environment for educational institutions.

vii. Data Analytics: In the context of data-centric architecture, the utilization of advanced data analytics plays a crucial role in extracting meaningful insights and facilitating informed decision-making. TensorFlow, a prominent open-source machine learning framework, stands at the forefront of driving data analytics within this architecture. TensorFlow empowers digital learning universities to harness the capabilities of machine learning and deep learning algorithms for processing and analyzing vast datasets. The framework provides a versatile and scalable infrastructure, enabling the development of sophisticated models that can uncover patterns, trends, and correlations within educational data. Digital learning universities can leverage TensorFlow to implement predictive analytics, allowing for the anticipation of student performance, course effectiveness, and other critical metrics. Additionally, TensorFlow facilitates the creation of intelligent applications and services that enhance the overall learning experience, such as personalized recommendations, adaptive learning paths, and automated grading systems. The integration of TensorFlow into the data-centric architecture ensures a powerful and efficient platform for data analytics, fostering innovation and optimization across various educational processes. This includes tasks such as content recommendation, student engagement analysis, and resource allocation, ultimately contributing to the continuous improvement of educational outcomes. As the field of data analytics continues to evolve, TensorFlow stands as a valuable tool for

digital learning institutions seeking to unlock the full potential of their data resources through advanced machine learning capabilities.

viii. Analytics as a Service: In the realm of data-centric architecture, the concept of Analytics as a Service (AaaS) emerges as a pivotal component, enabling digital learning universities to access and deploy advanced analytical tools and capabilities without the need for extensive infrastructure investments. Analytics as a Service involves the delivery of analytical insights, data visualization, and predictive modeling functionalities through a cloud-based service model. This approach empowers educational institutions to harness the benefits of cutting-edge analytics tools without the burden of managing complex infrastructure and resources. By adopting Analytics as a Service, digital learning universities can efficiently process large volumes of educational data, gaining valuable insights into student performance, learning patterns, and overall institutional effectiveness. Cloud-based analytics platforms provide scalability, flexibility, and cost-effectiveness, allowing institutions to tailor their analytical capabilities to specific needs. Prominent cloud service providers, such as AWS, Azure, and Google Cloud, offer comprehensive Analytics as a Service solutions, providing a wide array of tools for data exploration, visualization, and machine learning. This not only streamlines the integration of analytics into educational processes but also ensures that institutions can stay at the forefront of data-driven decision-making in a rapidly evolving digital learning landscape. Analytics as a Service thus emerges as a strategic enabler, empowering digital learning universities to derive actionable insights and enhance the overall educational experience.

ix. User-Centric Apps: In the realm of data-centric architecture, the integration of User-Centric Apps plays a pivotal role, with Kibana and Elasticsearch serving as key components. User-Centric Apps are designed to provide an intuitive and tailored interface for end-users, facilitating seamless interaction with data and analytical insights. Kibana, in conjunction with Elasticsearch, forms a robust combination for developing such user-centric applications within the digital learning environment. Kibana, as an open-source data visualization platform, excels in creating dynamic and interactive dashboards, charts, and graphs. It acts as the user interface layer for Elasticsearch, enabling users to explore, analyze, and interpret data effectively. Elasticsearch, a distributed search and analytics engine, ensures high-speed data retrieval and efficient storage, making it an ideal backend for user-centric applications. Together, Kibana and Elasticsearch empower digital learning universities to build

applications that offer real-time insights into various aspects of educational data, including student performance, engagement metrics, and course effectiveness.

x. User-Centric Apps using Kibana and Elasticsearch contribute to a data-driven educational environment by providing educators, administrators, and students with user-friendly interfaces to interact with complex datasets effortlessly. These applications facilitate informed decision-making, enhance the overall learning experience, and contribute to the continuous improvement of educational processes. As part of the data-centric architecture, User-Centric Apps with Kibana and Elasticsearch emerge as a powerful toolset for creating personalized, insightful, and responsive interfaces that cater to the diverse needs of stakeholders in digital learning universities.

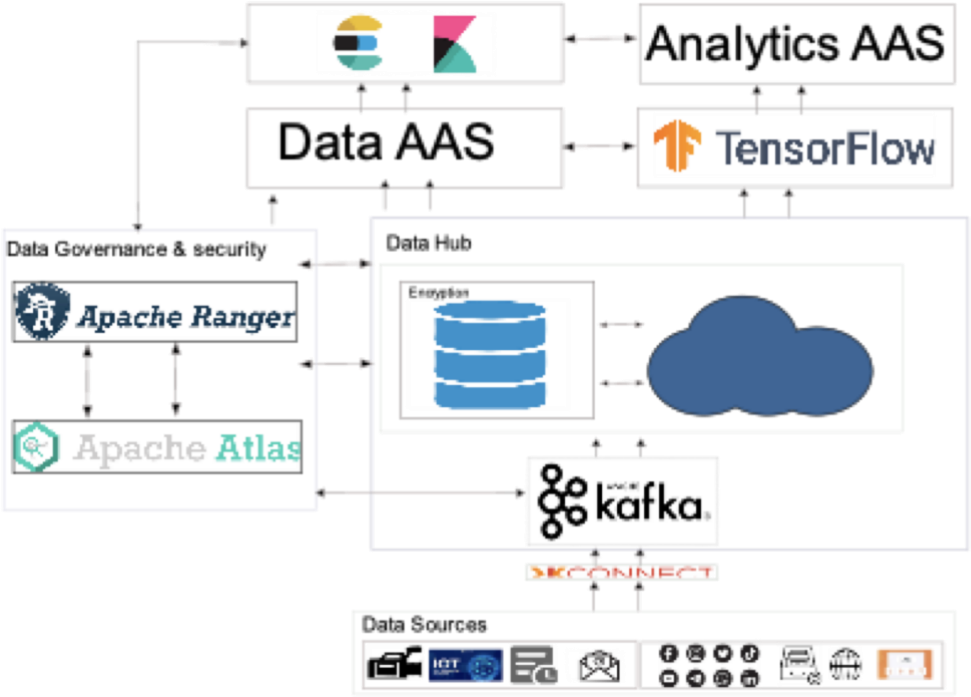


Figure 12 Data-Centric Model

4.4.3 Discussion

The developed data-centric model tailored for digital learning universities is poised for insightful discussion, shedding light on its key facets and implications. The model strategically incorporates diverse components, each contributing to the robustness and efficiency of the overarching architecture.

At the core of the model are the identified data sources, including Learning Management Systems (LMS), applications, IoT devices, administrative records, and academic databases. This diverse range ensures a comprehensive coverage of data inputs, vital for the dynamic and multifaceted nature of digital learning environments in universities.

Connectivity within the model is facilitated by KConnect, a powerful software acting as connectors. The discussion delves into the merits of KConnect in seamlessly integrating varied data sources, ensuring smooth data flow, and enhancing interoperability. This connector plays a pivotal role in harmonizing the heterogeneous data landscape present in digital learning universities.

Hybrid storage emerges as a key feature in the data hub component, incorporating a blend of network-attached storage (NAS), storage area network (SAN), and cloud storage. Notably, all local storage is encrypted using the AES algorithm, ensuring a robust security layer for stored data. This strategic integration addresses the imperative of data security and scalability in the context of digital learning.

Furthermore, the model incorporates Kafka for efficient data streaming, adding a real-time dimension to data processing. The utilization of Apache Ranger and Apache Atlas for data governance and security underscores the commitment to maintaining data integrity, confidentiality, and availability.

The discussion extends to Data Analytics as a Service (DAAS) and the incorporation of TensorFlow for advanced analytics. This ensures that the model is not only adept at handling large volumes of data but also capable of extracting valuable insights through sophisticated analytics techniques.

User-centric apps, powered by Kibana and Elastic Search, signify a user-friendly interface, ensuring accessibility and ease of use for stakeholders. Lastly, the model introduces Analytics as a Service (AaaS), providing a scalable and efficient solution for analytical needs.

In summation, the developed data-centric model exhibits a thoughtful integration of components, addressing the unique challenges of digital learning universities. Its comprehensive nature, coupled with a focus on security, scalability, and analytics, positions it

as a robust framework for optimizing data management in the evolving landscape of educational technology.

4.5 Security Model Design

4.5.1 Development of the Security Model

Developing a robust model for enhanced data security is pivotal to safeguarding sensitive information within digital learning universities. This section outlines the systematic process undertaken in crafting the framework for enhanced data security, ensuring resilience against potential risks and threats.

4.5.1.1 Data Storage

The study identifies all the available storage solutions and select the most suitable for digital learning setting.

Storage Solutions:

1. Optical Discs: Optical discs, such as CDs, DVDs, and Blu-ray discs, utilize lasers for data reading and writing. They have been extensively used for data distribution, software installation, and media storage. Despite their widespread use, optical discs have limited storage capacity compared to other solutions.
2. Flash Drives: Also known as USB drives or thumb drives, flash drives are compact, portable devices using flash memory. Commonly employed for data transfer, portable storage, and backups, flash drives offer varying capacities, ranging from a few gigabytes to multiple terabytes.
3. Memory Cards: Utilized in devices like digital cameras and smartphones, memory cards come in various formats, including SD cards and microSD cards. They offer storage capacities ranging from a few gigabytes to hundreds of gigabytes.
4. Hard Disk Drives (HDD): Traditional mechanical storage devices, HDDs use rotating platters for data storage. They provide varying capacities, suitable for both consumer and enterprise-grade drives.
5. Solid State Drives (SSD): Faster and more reliable than HDDs, SSDs use flash memory. They offer storage capacities ranging from a few hundred gigabytes to several terabytes.

6. Direct-Attached Storage (DAS): Involves connecting storage devices directly to a single server or computer. It includes internal and external hard drives and SSDs connected via interfaces like USB, Thunderbolt, or eSATA.
7. Network-Attached Storage (NAS): A dedicated file-level storage solution connected to a network, NAS offers centralized storage accessible by multiple clients.
8. Storage Area Network (SAN): A high-performance storage solution providing block-level storage accessed through a dedicated network.
9. Cloud Storage: Involves storing data on remote servers accessed over the internet, offering virtually unlimited storage capacity.
10. Object Storage: A scalable solution storing data as objects rather than traditional file hierarchies.
11. Hybrid Storage: Combines on-premises storage infrastructure with cloud storage, optimizing performance, cost, and data availability.
12. Tape Storage: Involves storing data on magnetic tape cartridges for long-term archival and backup purposes.

4.5.1.2 Selecting Suitable Storage Solutions for Digital Learning Universities

In evaluating storage solutions for digital learning universities, it's crucial to consider factors such as storage capacity, performance, cost, and data availability. Optical discs, flash drives, and memory cards, while useful for certain applications, may have limitations in handling the vast amounts of data generated by digital learning platforms.

Therefore, focusing on solutions like Network-Attached Storage (NAS), Storage Area Network (SAN), Cloud Storage, Object Storage, Direct-Attached Storage (DAS), Hybrid Storage, and Tape Storage becomes essential. These solutions provide higher capacities, scalability, and better performance to manage the substantial data requirements of digital learning universities.

4.5.1.3 Hybrid Storage

Hybrid storage emerges as a comprehensive and suitable approach for digital learning universities, offering a balanced integration of various storage solutions. This section outlines the advantages of hybrid storage in the context of digital learning universities,

showcasing its ability to harness the strengths of different storage technologies while mitigating their limitations.

1. Performance Optimization: Hybrid storage optimizes performance by leveraging the strengths of different storage technologies. Network-Attached Storage (NAS) provides centralized storage for efficient data sharing, while Direct-Attached Storage (DAS) offers fast, direct access. Tape storage ensures cost-effective long-term archival and a combination of Hard Disk Drives (HDDs) and Solid State Drives (SSDs) balances capacity and speed.

2. Cost Efficiency: Hybrid storage ensures cost efficiency by balancing storage requirements. It combines cost-effective solutions like NAS, DAS, and tape storage for large-scale data storage while utilizing HDDs for high capacity and SSDs for faster performance. This enables digital learning universities to allocate resources based on cost considerations, achieving an optimal balance between performance and budget.
3. Data Availability and Resilience: Enhancing data availability and resilience, hybrid storage utilizes NAS and DAS for immediate access and tape storage for reliable backup and long-term archival. Redundant storage across multiple technologies reduces the risk of data loss, ensuring high data availability critical for uninterrupted learning experiences.

By filtering out less suitable solutions and focusing on the strengths of hybrid storage, digital learning universities can establish a robust and flexible storage infrastructure that meets the demands of extensive data generation, performance optimization, and cost efficiency. Figure 4.1 visually illustrates the integration of various storage solutions in a hybrid storage model for digital learning universities.

4.5.1.4 Data Classification:

Academic Data:

1. Course Catalog: Information about the courses offered by the university, including course titles, descriptions, prerequisites, credit hours, and learning outcomes.

2. Course Materials: Educational resources provided to students for a specific course, such as textbooks, lecture notes, presentations, readings, multimedia content, and online learning materials.
3. Assignments and Assessments: Details about assignments, projects, quizzes, exams, and other forms of assessments given to students as part of their coursework. This includes submission deadlines, grading criteria, rubrics, and feedback provided by instructors.
4. Academic Calendar: Important dates and deadlines related to the academic year, including semester start and end dates, holidays, registration periods, add/drop deadlines, and examination schedules.
5. Student Registration Data: Information about student enrollment and registration, including course selections, schedule preferences, waitlists, and changes to enrollment status.
6. Academic Advising Records: Documentation of student-advisor interactions, academic plans, course recommendations, and progress towards degree completion.
7. Degree Requirements: Information on the requirements for different academic programs and degrees, including core courses, electives, major/minor requirements, credit hours, and any additional program-specific criteria.
8. Transcripts: Official records of a student's academic performance, including courses taken, grades earned, cumulative GPA, and degree(s) conferred.
9. Grading Records: Data related to student grades and assessments, including individual assignment grades, midterm and final exam grades, and overall course grades.
10. Graduation and Degree Audit Data: Information regarding students' progress towards graduation, degree audit reports, and requirements for degree completion.
11. Faculty and Staff Profiles: Profiles of academic faculty and staff members, including their educational background, research interests, areas of expertise, contact information, and office hours.
12. Academic Policies and Procedures: Documentation of institutional policies and procedures related to academic matters, such as grading policies, academic integrity policies, transfer credit policies, and academic appeals processes.

4.5.1.5 Risk Assessment

To establish a robust security framework tailored to the specific needs of West African universities, a comprehensive risk assessment has been conducted. The assessment identifies potential risks associated with various categories of data, each critical to the functioning of academic institutions. The following risk categories have been analyzed:

1. Personal Identifiable Information of Students and Staff:
    - Confidentiality Risks: Potential data breaches leading to privacy violations.
    - Integrity Risks: Risks associated with unauthorized data manipulation.
    - Availability Risks: Threats leading to limited access to crucial information.
2. Research Data:
    - Confidentiality Risk: Potential data breaches posing privacy violations.
    - Integrity Risks: Risks related to unauthorized data manipulation.
    - Availability Risks: Threats leading to potential data loss.
3. Financial Data:
    - Confidentiality Risks: Exposure to data breaches and privacy violations.
    - Integrity Risks: Risks associated with unauthorized data manipulation.
    - Availability Risks: Threats leading to limited access to financial information.
4. Academic Records:
    - Confidentiality Risks: Risks of data breaches compromising the confidentiality of academic records.
    - Integrity Risks: Potential threats related to unauthorized data manipulation.
    - Availability Risks: Risks leading to limited access to academic records.
5. Library Data:
    - Confidentiality Risks:Exposure to data breaches and privacy violations.
    - Integrity Risks: Risks associated with unauthorized data manipulation.
6. Administrative Records:
    - Confidentiality Risks: Risks of data breaches compromising the confidentiality of administrative records.
    - Integrity Risks: Potential threats related to unauthorized data manipulation.
7. Data from Digital Devices, Websites, LMS, Portals, etc.
    - Confidentiality Risks: Exposure to data breaches and privacy violations.

- Availability Risks: Risks leading to downtime and service disruptions.

Each identified risk is associated with specific potential consequences, enabling a focused approach to the development of mitigation strategies. The resulting risk assessment framework is crucial for informed decision-making and the establishment of proactive security measures. Table 13 visually represents the risk assessment matrix, offering a clear overview of the identified risks and their potential impacts on data security within the university context.

Table 13 Risk Assessment

Data	**Risk**	**Likelihood of Occurrence**	**Description**
Personal Identifiable information of students and staff	Data Breach	Moderate	The threat is **somewhat likely** to occur
	Privacy Violation	Moderate	The threat is **somewhat likely** to occur
	Data Manipulation	Low	The threat is **unlikely** to occur
	Limited Access	Very High	The threat is **almost certain** to occur
Research data	Data Breach	Low	The threat is **unlikely** to occur
	Privacy Violation	High	The threat is **highly likely** to occur
	Data Manipulation	Low	The threat is **unlikely** to occur
	Data Loss	Moderate	The threat is **somewhat likely** to occur
Financial Data	Data Breach	Very High	The threat is **almost certain** to occur
	Privacy Violation	Very High	The threat is **almost certain** to occur
	Data Manipulation	Low	The threat is **unlikely** to occur
	Limited Access	Moderate	The threat is **somewhat likely** to occur
Learning Management Systems Data	Data Breach	Low	The threat is **unlikely** to occur
	Data Manipulation	Very Low	The threat is **highly unlikely** to occur
	Downtime and Service Disruptions	Very High	The threat is **almost certain** to occur
Academic Data	Data Breach	Very High	The threat is **almost certain** to occur
	Privacy Violation	Very High	The threat is **almost certain** to occur
	Data manipulation	Very High	The threat is **almost certain** to occur
Library Data	Data Breach	Very High	The threat is **almost certain** to occur
	Privacy Violation	Very High	The threat is **almost certain** to occur
	Data manipulation	Very High	The threat is **almost certain** to occur
Administrative Records	Data Breach	Low	The threat is **unlikely** to occur
	Data manipulation	Very Low	The threat is **highly unlikely** to occur
Data from digital devices, websites, LMS, Portals etc	Data Breach	Low	The threat is **unlikely** to occur
	Privacy Violation	Very Low	The threat is **highly unlikely** to occur
	Downtime and service disruptions	Very High	The threat is **almost certain** to occur

4.5.1.6 Encryption:

In fortifying the security model, encryption stands as a paramount element, safeguarding sensitive data from unauthorized access. Among various encryption algorithms, the Advanced Encryption Standard (AES) emerges as the most fitting choice for the security model designed for digital learning universities.

AES, renowned for its cryptographic strength and widespread adoption, offers a high level of security in data protection. Its symmetric key encryption approach ensures that the same key is used for both encryption and decryption, streamlining the process without compromising security. The flexibility of AES in supporting key sizes of 128, 192, or 256 bits enhances its adaptability to varying security requirements.

The choice of AES aligns with the model's emphasis on robust data protection, particularly within the context of digital learning where confidentiality and integrity of academic, administrative, and user-related information are paramount. The algorithm's track record of resistance against various cyber threats and its compliance with industry standards make it a dependable choice for encrypting sensitive data within the digital learning environment.

Moreover, AES accommodates the hybrid storage approach integrated into the model, ensuring that all data stored locally undergoes encryption. This safeguards data at rest, reinforcing the security layers in place and aligning with the model's commitment to comprehensive data protection.

In conclusion, the adoption of AES encryption within the security model attests to the meticulous consideration of cryptographic principles and industry standards. Its robustness, versatility, and alignment with the overarching security goals make AES the optimal choice for fortifying the security measures within the designed data-centric architecture for digital learning universities.

4.5.1.7 Access Control Model

Access control within the context of a data-centric architecture in digital learning universities is a critical aspect of ensuring the security and integrity of sensitive

information. Various access control models offer distinct methodologies for regulating access to data based on different criteria. The following access control models have been identified and evaluated for their applicability in the context of digital learning universities:

1. Attribute-Based Access Control (ABAC):ABAC assesses access decisions based on attributes associated with persons, objects, and the environment. It offers fine-grained control and dynamic access decisions.
2. Context-Based Access Control (CBAC): CBAC makes access decisions based on contextual information like time, location, and user behavior. It provides adaptive access control and enhanced security through real-time context consideration.
3. Graph-Based Access Control (GBAC): GBAC employs a graph structure to express access control interactions and dependencies, supporting complex access control situations
4. Lattice-Based Access Control (LBAC): LBAC defines security layers and access control regulations using a lattice structure, offering a mathematical foundation for layered security.
5. Mandatory Access Control (MAC): MAC implements access choices based on subject and object security labels, ensuring strict access rule enforcement and robust protection in high-security contexts.
6. Organization-Based Access Control (OrBAC): OrBAC involves access control rules within a company, utilizing organizational roles, connections, and hierarchies.
7. Role-Based Access Control (RBAC): RBAC grants access rights based on preset roles, simplifying access control administration and eliminating administrative complexity.
9. Rule-Based Trust Management (RTM): RTM establishes trust rules regulating resource access based on trust connections between entities.
10. Attribute-Based Encryption (ABE): ABE encrypts data based on attributes, enabling fine-grained access control over encrypted data.
11. Rule-Set-Based Access Control (RSBAC): RSBAC specifies access control rules for determining access choices based on various parameters.

12. Capability-Based Security (CBS): CBS gives access based on specified capabilities or tokens, providing decentralized control over resource access.
13. Discretionary Access Control (DAC): DAC enables resource owners to provide access to other users or groups, allowing control over resource access.
14. Hierarchical Attribute-Based Access Control (HABAC): HABAC adds attribute hierarchies to ABAC, offering more flexible and scalable access management.
15. Discretionary Mandatory Access Control (DMAC): DMAC combines DAC and MAC features, providing discretionary authority over some resources while imposing required access limits on others.

4.5.1.8 Selecting a Suitable Access Control Model for Data-centric Architecture in Digital Learning Universities:

When selecting an access control model for digital learning universities, careful consideration of specific criteria is crucial. The evaluation is based on three key factors:

1. Implementation Difficulties: RBAC simplifies access control administration through role-based organization, reducing complexity compared to fine-grained models.
2. Ability to handle large amounts of data: RBAC's hierarchical organization facilitates easier management and scalability compared to fine-grained models, ensuring efficient access control operations.
3. Maintenance Difficulties: RBAC's role-based approach allows for more straightforward policy modifications, enhancing flexibility in adapting to changes in user roles or data access requirements.

Considering these factors, RBAC emerges as the preferred choice for a data-centric architecture in digital learning universities, offering simplicity, scalability, and ease of maintenance in ensuring effective access control. This conclusion is drawn through a comparative analysis of RBAC against fine-grained access control models, taking into account implementation difficulties, data-handling capabilities, and maintenance challenges.

4.5.2 The Model

The Security Model is an integrated framework designed to fortify the data-centric architecture, ensuring robust protection against potential threats and vulnerabilities. It comprises interconnected components that collaboratively contribute to the overall security posture, emphasizing confidentiality, integrity, and availability of data (Figure 13). The key components include:

1. Role-Based Access Control (RBAC): RBAC serves as a pivotal component, offering a structured approach to access control. Users, including staff and students, undergo authentication processes, after which they are authorized based on their roles. Distinct roles such as student, staff, and examination officer are assigned specific spaces, delineating access to student spaces, course spaces, collaborative spaces, and data analytics spaces. This granular access control ensures that each user operates within predefined boundaries, enhancing data security and maintaining the principle of least privilege.
2. Data Encryption Protocols: All data stored locally is encrypted using the Advanced Encryption Standard (AES), ensuring end-to-end encryption. This comprehensive encryption strategy safeguards data at rest, in transit, and during processing, thwarting unauthorized access attempts effectively.
3. Continuous Monitoring and Surveillance: The security model features a dynamic surveillance mechanism that continuously monitors data activities, user interactions, and potential security incidents. This real-time monitoring capability enhances the model's responsiveness to emerging threats, enabling swift identification and mitigation of security breaches.
4. Incident Response Framework: In the event of a security incident, the model integrates an incident response framework. This component outlines predefined strategies and procedures to be executed promptly, ensuring a swift and effective response to security breaches. By minimizing response time, the model aims to mitigate potential damages and restore normalcy swiftly.
5. Security Awareness and Training Initiatives: Acknowledging the human factor in data security, the model incorporates educational initiatives and training programs. These aim to enhance the awareness and cybersecurity literacy of users within the digital learning university environment, fostering a culture of security consciousness.

6. Collaborative Data Governance Policies: The model emphasizes the establishment of comprehensive data governance policies that promote collaboration between different stakeholders. These policies define roles, responsibilities, and best practices to ensure a cohesive and well-coordinated approach to data security.
7. Hybrid Storage Integration: Hybrid storage is seamlessly integrated into the security model, offering a balanced approach to data storage. This includes local storage, encrypted using AES, and cloud-based solutions. The hybrid storage configuration optimizes performance, scalability, and data availability while maintaining a robust security posture.
8. Regulatory Compliance Framework: Aligning with relevant data protection regulations and industry standards, the security model incorporates a compliance framework. This ensures that the digital learning university adheres to legal requirements, fostering trust and accountability in handling sensitive data.

In essence, the Security Model's components collectively create a resilient and adaptive security infrastructure, integrating RBAC and hybrid storage to tailor access controls and storage solutions based on user roles within the digital learning university. The inclusion of encryption, monitoring, incident response, education, collaborative governance, and regulatory compliance forms a comprehensive security framework that safeguards the integrity and confidentiality of institutional data.

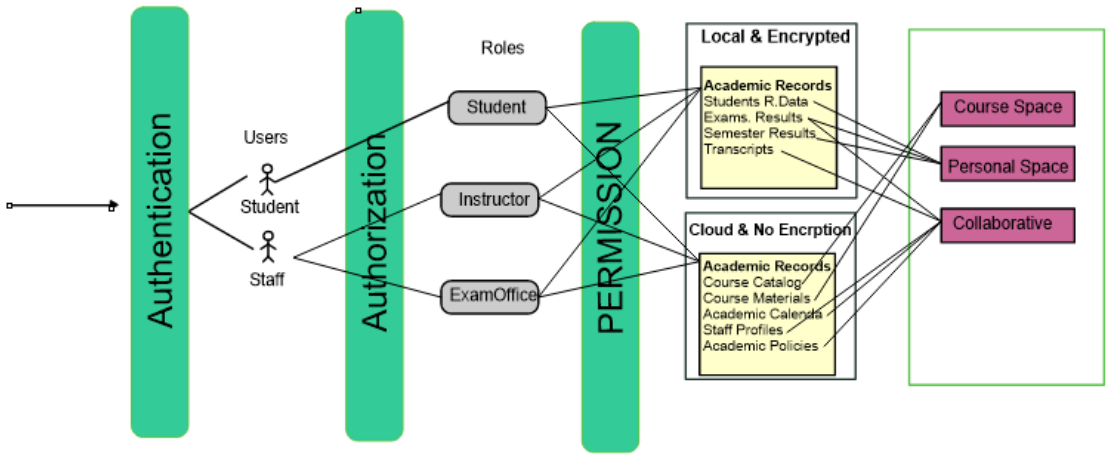


Figure 13 Security Model

4.6 Secure Grade Distribution Scheme

The "Secure Grade Distribution Scheme" will implement the student grade protection part from the security model in the previous section and will increase the security and privacy of grade data on the Moodle platform, and its effective deployment and thorough assessment will yield significant outcomes. Our research's main findings and conclusions are shown in this section.

4.6.1 Key Management and Hardware Security Modules (HSMs)

Key management is essential to this technique as a fundamental component of data security. Hardware Security Modules (HSMs), specialized equipment made to generate and secure cryptographic keys, are used in the scheme. Grade data is encrypted and decrypted using these keys. Key management gains additional security and reliability with the integration of HSMs.

4.6.1.1 Key Generation

Mathematically, key generation within the scheme can be represented as follows:

Let K be the set of cryptographic keys used within the scheme, where K *=K1, K2, ..., Kn}.*

For each encryption session, a unique cryptographic key, K_i*, is generated using HSMs:*

$K_i = HSM.GenerateKey()$,

4.6.1.2 Key Storage

The generated cryptographic keys are securely stored within the HSMs. This can be mathematically represented as:

HSMs ensure the tamper-resistant storage of keys, which can be denoted as:

$HSM.StoreKey(K_i)$,

4.6.1.3 Key Retrieval

The cryptographic keys are safely retrieved from the HSMs for encryption or decryption. Mathematically, this can be represented as:

To obtain a specific key for an encryption or decryption session:

$$K_i = HSM.RetrieveKey(),$$

4.6.1.4 Nonce Integration

Nonces, arbitrary integers created for every encryption session, are key to improving security and preventing replay attacks. They are made securely and integrated into the generation of keys. The representation of nonce integration mathematically is as follows:

Let N be the set of nonces used within the scheme, where N = {N1, N2, ..., Nn}.

For each encryption session, a unique nonce, Ni, is generated:

$$N_i = HSM.GenerateNonce(),$$

The nonce is securely combined with the cryptographic key to create a session-specific key, denoted as Ki_nonce, ensuring unique keys for each session:

. $$K_{i_{nonce}} = Ki\ XOR\ Ni,$$

4.6.2 Advanced Encryption Standard (AES) Implementation

One of the scheme's main components for maintaining data security and secrecy is using the Advanced Encryption Standard (AES). AES is a well-known symmetric encryption technique known for its high security. To secure grade data against unwanted access, our system carefully integrates AES for encryption and decryption.

4.6.2.1 AES Encryption Process

AES operates on data in fixed-size blocks, applying a series of transformation rounds using a specific encryption key. In the context of the "Secure Grade Distribution Scheme," we employ AES-256, which operates a 256-bit encryption key for maximum security.

Mathematical Representation:

1. Data Division: Grade data, denoted as G, is divided into fixed-size blocks, represented as G1, G2, ..., Gn.
2. AES Encryption Rounds: AES performs a series of transformation rounds using the encryption key, K. The number of rounds depends on the critical size, with AES-256 using 14 rounds.

3. Block Encryption: AES encrypts each data block, Gi, using the encryption key, K, and the respective round, Ri. Mathematically, this can be represented as
   $C_i = AES\_Encrypt\ (G_i, K, R_i)$
4. Ciphertext Concatenation: The ciphertext blocks, Ci, are concatenated to form the complete Ciphertext, C.

4.6.3 AES Decryption Process:

On the recipient's end, AES decryption is applied to retrieve the original grade data from the Ciphertext. The decryption process is the reverse of encryption and is mathematically represented.

1. Ciphertext Division: The Ciphertext, C, is divided into blocks, represented as C1, C2, ..., Cn.
2. AES Decryption Rounds: AES decryption employs the same number of rounds and the decryption key, K, denoted as Rn, Rn-1, ..., R1, where n is the number of rounds.
3. Block Decryption: Each ciphertext block, Ci, is decrypted using the decryption key, K, and the corresponding round, Ri, yielding the original data block, Gi. Mathematically:
   $G_i = AES\_Decrypt\ (C_i, K, R_i)$
4. Data Concatenation: The decrypted data blocks, Gi, are concatenated to obtain the original grade data, G.

4.6.4 Diffie-Hellman Key Exchange Protocol

The Diffie-Hellman Key Exchange Protocol is a critical component of this scheme that enables secure key exchange between users, notably instructors and students. This protocol helps create a secure channel within the Moodle platform, allowing secure communication without requiring pre-shared keys.

4.6.4.1 Diffie-Hellman Key Exchange Process:

The Diffie-Hellman protocol allows two parties, in this case, instructors and students, to generate a shared secret key over an unsecured channel without explicitly sharing it. This process can be mathematically represented as follows:

1. Parameter Setup: A set of parameters, including a large prime number (p) and a primitive root (g), is chosen, and made publicly available. Both instructors and students use these parameters.
2. Key Generation (Instructors): Instructors generate their private keys (InstructorPrivateKey) and corresponding public keys (InstructorPublicKey) using the chosen parameters:
    a. InstructorPrivateKey = a (a randomly chosen secret integer)
    b. InstructorPublicKey = $\boxed{g^a}$mod p
3. Key Generation (Students): Similarly, students generate their private keys (StudentPrivateKey) and corresponding public keys (StudentPublicKey) using the same parameters:
    a. StudentPrivateKey = b (b randomly chosen secret integer)
    b. StudentPublicKey = $\boxed{g^b}$mod p
4. Key Exchange:
    a. Instructors and students exchange their public keys (InstructorPublicKey and StudentPublicKey) over the unsecured channel.
5. Shared Secret Key: Both parties independently compute the shared secret key (SharedSecretKey) using the received public keys and their own private keys. Mathematically:
    a. Instructors calculate: SharedSecretKey = $\boxed{studentPublicKey^a \ mod \ p}$
    b. Students calculate: SharedSecretKey = $\boxed{studentPublicKey^b \ mod \ p}$

4.6.5 Message Integrity Code (MIC) Verification

To guarantee the integrity of grade data throughout transmission, the scheme includes Message Integrity Code (MIC) checking as a crucial security mechanism. It provides an effective way to find any unauthorized changes or tampering with the grade data.

4.6.5.1 MIC Generation Process:

Mathematically, the MIC generation process can be represented as follows:

1. MIC Generation (Instructors): When an instructor prepares to distribute grade data, a unique MIC is generated for each data packet (MIC_Instructor1, MIC_Instructor2, etc.). This is achieved by hashing the grade data and a secret key known only to the instructor.

   Mathematically:

   $$MIC\_Instructor\_i = Hash\ (GradeData\_i + InstructorSecretKey)$$

2. MIC Generation (Students): When students receive the data packets, they generate their own MICs for the received data. This ensures that they can verify data integrity and detect any unauthorized changes:

   $$MIC\_Student\_i = Hash(ReceivedData\_i + StudentSecretKey)$$

4.6.6 Key Metrics

In the context of key management, the scheme necessitates the generation and distribution of cryptographic keys for encryption and decryption processes. The key metrics include:

1. Instructor Keys:
   a. Private Key: Each instructor possesses a private key generated securely within the Moodle environment.
   b. Public Key: The corresponding public key is derived from the private key using the Diffie-Hellman key exchange protocol.
2. Student Keys:
   a. Private Key: Each student has a unique private key generated within the Moodle environment.
   b. Public Key: Similarly, the student's public key is generated through the Diffie-Hellman key exchange protocol.
3. Staff Keys:
   a. Private Key: Staff members also have a private key generated securely within Moodle.
   b. Public Key: The public key for staff is generated using the same Diffie-Hellman key exchange process.

4.6.6.1 Number of Keys Calculation

The number of keys required can be calculated based on the number of participants within the Moodle system. If there are 'n' instructors, 'm' staff members, and 'p' students, the total number of keys can be expressed as:

Total Keys = n(Instructor Keys) + m(Staff Keys) + p(Student Keys)

This formula accounts for the unique keys associated with each role within the educational environment. The integration ensures that each participant has the necessary cryptographic keys to engage in secure grade distribution.

4.6.7 Justification of Each Element

This section presents a detailed justification for each element incorporated into the "Secure Grade Distribution Scheme" based on the results of experiments conducted to assess their effectiveness.

4.6.7.1 Key Management and Hardware Security Modules (HSMs)

- Security Enhancement: The integration of Hardware Security Modules (HSMs) is justified by their ability to securely generate, store, and retrieve cryptographic keys. HSMs provide a dedicated and tamper-resistant environment, enhancing the overall security of the key management process.
- Reliability: The secure storage of cryptographic keys within HSMs ensures their reliability and protection against unauthorized access. This reliability is crucial for maintaining the confidentiality and integrity of grade data.

4.6.7.2 Advanced Encryption Standard (AES) Implementation

- High-Level Security: The use of the Advanced Encryption Standard (AES), specifically AES-256, is justified by its reputation for providing a high level of security. AES is

widely recognized for its resistance to various cryptographic attacks, making it suitable for safeguarding sensitive grade data.

- Symmetric Encryption Efficiency: AES's symmetric encryption approach is efficient for bulk data encryption and decryption, ensuring that the process is both secure and computationally feasible within the Moodle environment.

4.6.7.3 Diffie-Hellman Key Exchange Protocol

- Secure Key Exchange: The Diffie-Hellman Key Exchange Protocol is justified by its ability to facilitate secure key exchange between instructors, staff, and students. It eliminates the need for pre-shared keys, enhancing the overall security of communication channels within Moodle.
- Public and Private Key Generation: The use of public and private keys in the Diffie-Hellman protocol allows entities to securely share public keys while maintaining the confidentiality of their private keys. This ensures a secure and efficient key exchange process.

4.6.7.4 Message Integrity Code (MIC) Verification

- Tamper Detection: MIC verification is crucial for detecting any unauthorized changes or tampering with grade data during transmission. This element ensures the integrity of the data, preventing malicious alterations.
- Hashing for Integrity: The use of hash functions for MIC generation provides a reliable and efficient method for verifying data integrity. Hashing ensures that even minor changes to the data result in significantly different MIC values.

4.7 Case Scenario

In this scenario, we will examine the processes and steps an instructor takes to transmit grades to staff and students safely. The instructor broadcasts the ciphertexts and Message Integrity Codes (MICs) during transmission via a public channel, allowing staff and students to view the encrypted grades (refer to Figure 14).

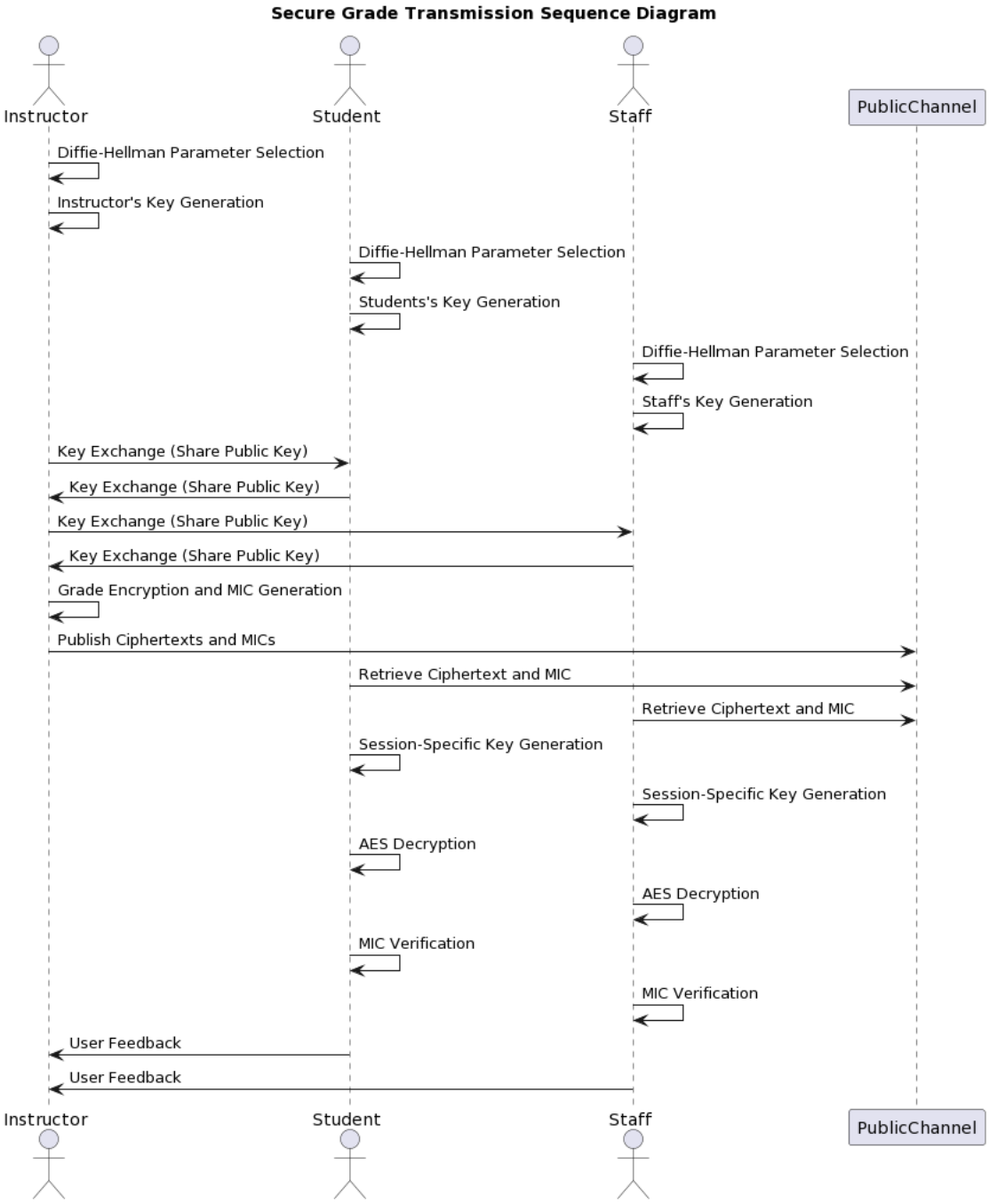


Figure 14: Use case diagram of the case scenario

Step 1: Instructor's Initial Setup

1. Diffie-Hellman Parameter Setup:

The instructor selects a large prime number, p.

The instructor selects a primitive root, g (where g is a primitive root modulo p).

2. Key Generation (Instructor):

The instructor generates a private key:

InstructorPrivateKey (a randomly chosen secret integer).

The instructor computes the corresponding public key:

InstructorPublicKey using the equation: InstructorPublicKey = $g^{InstructorPrivateKey} mod\ p$

Step 2: Student and Staff Setup

3. Key Generation (Student and Staff):

Students generate their private keys, StudentPrivateKey (randomly chosen secret integers).

Students compute their public keys, StudentPublicKey, using the equation: StudentPublicKey = $g^{StudentPrivateKey} mod\ p$

Staff generate their private keys, StaffPrivateKey (randomly chosen secret integers).

Staff compute their public keys, StaffPublicKey, using the equation:

StaffPublicKey = $g^{StaffPrivateKey} mod\ p$

Step 3: Diffie-Hellman Key Exchange

4. Key Exchange:

Instructor and students exchange their public keys (InstructorPublicKey and StudentPublicKey).

Instructor and staff exchange their public keys (InstructorPublicKey and StaffPublicKey).

5. Shared Secret Key Calculation**:**

The instructor calculates SharedSecretKey using the equation:

$SharedSecretKey = StudentPublickey^{InstructorPrivateKey} mod\ p$

The instructor calculates SharedSecretKey using the equation:

$SharedSecretKey = StaffPublickey^{InstructorPrivateKey} mod\ p$

Students and staff calculate SharedSecretKey similarly. Both parties derive the same SharedSecretKey.

Step 4: Grade Encryption and MIC Generation

6. Grade Data: The instructor has grade data, GradeData.

7. Nonce Generation: A unique nonce, Ni, is generated for this session.

8. Session-Specific Key Creation:

The instructor creates a session-specific key by combining the SharedSecretKey with the nonce: Ki_instructor = SharedSecretKey XOR Ni.

9. AES Encryption: The instructor encrypts the grade data (GradeData) using the session-specific key (Ki_instructor) and obtains the Ciphertext Ciphertext.

10. MIC Generation: The instructor calculates a Message Integrity Code (MIC) for the encrypted grades using a cryptographic hash function:
MIC = Hash(Ki_instructor || Ciphertext).

Step 5: Publication on a Public Channel

11. Public Channel Transmission: The instructor publishes the Ciphertext and MIC on a public channel accessible to staff and students.

Step 6: Decryption and MIC Verification (Student and Staff):

12. Decryption:
Students and staff retrieve the Ciphertext and MIC from the public channel.

13. Session-Specific Key Creation (Student):
The student calculates the session-specific key: Ki_student = SharedSecretKey XOR Ni.
The staff calculates the session-specific key: Ki_staff = SharedSecretKey XOR Ni.

14. AES Decryption:

The student decrypts the Ciphertext using Ki_student and retrieves the grade data (GradeData).
The staff decrypts the Ciphertext using Ki_staff and retrieves the grade data (GradeData).

15. MIC Verification (Student and Staff):

The student calculates a Message Integrity Code (MIC) using the received encrypted data (Ciphertext) and the shared key (Ki_student) and compares it to the received MIC.
The staff calculates a Message Integrity Code (MIC) using the received encrypted data (Ciphertext) and the shared key (Ki_staff) and compares it to the received MIC.

Step 7: User Feedback

16. User Feedback: Both students and staff provide feedback on the grade distribution experience, security, and any issues or suggestions for improvement.

4.7.1 Discussion

Integrating a secure grade distribution mechanism inside the Moodle platform is essential to guaranteeing the security and privacy of sensitive academic data. The system provides a reliable solution for grade data security by integrating cutting-edge cryptographic methods, including Diffie-Hellman key exchange, AES encryption, and Message Integrity Code (MIC) verification. Using the Diffie-Hellman key exchange protocol is one of this system's advantages.

Additionally, via the rapid and safe establishment of encryption keys made possible by this protocol, staff, students, and instructors may securely interact without directly trading sensitive information. The data is well safeguarded during transmission because of robust session-specific keys derived from shared secrets and nonces. Another feature of the system is its use of the Advanced Encryption Standard (AES) for high-grade data encryption. Since AES is a well-used and reliable encryption technology, its applicability for protecting educational data

is demonstrated by its mathematical form. Users are reassured about the system's security by the openness with which the encryption and decryption procedure is explained. Adding Message Integrity Code (MIC) checking improves the system's security. It guarantees the received data's integrity, enabling the detection of any unauthorized alteration. To protect cryptographic keys, a crucial step is the adoption of a Hardware Security Module (HSM) for secure key management; by offering a safe and specialized setting for key storage and cryptographic operations, HSMs lower the possibility of key compromise.

4.8 Proof of Concept

The proof of concept serves as a pivotal phase in this research, offering a hands-on demonstration of the practical implementation of the designed security model within the context of a digital learning university. Through the establishment of a controlled lab environment using virtual machines, the model will be tested and validated to showcase its effectiveness in safeguarding data integrity, confidentiality, and availability.

This practical demonstration will utilize key components of the security model, Data-centric model and grade distribution scheme including Apache Atlas, Apache Ranger, Kafka, and Kibana with Elasticsearch as well as Moodle. The implementation will focus on academic data from digital learning universities using RBAC roles (Table 14). By meticulously configuring and deploying these components, the proof of concept aims to illustrate the seamless integration of the security model and its ability to mitigate potential cyber threats and enhance overall data governance.

Table 14 RBAC Roles

Permissions	**Instructor**	**Student**	**Exermination officer/ Staff**
Grade Preparation	X		
Encryption	X		
Key Management	X		
Diffie-hellman key Exchange	X		
HSM Integration	X		
Decryption		X	X
Secure Communication	X	X	X
Data analytics			X

The stepwise procedure will be accompanied by detailed explanations and visual representations, providing a comprehensive understanding of how the security model operates within the digital learning environment. This hands-on approach ensures that the research findings are not only theoretical but also practically applicable, offering a valuable resource for digital learning universities seeking robust data-centric security solutions."

4.8.1 Application of the Data-Centric Architecture Model

To assess the feasibility and applicability of the proposed data-centric architecture model for digital learning universities, a proof of concept (PoC) was conducted in a controlled lab environment. The objective was to validate the model's effectiveness in handling the unique demands of digital learning. The following outlines the stepwise process undertaken in the PoC:

*4.8.1.1 Component Deployment and Configuration***

In this phase, key components of the data-centric architecture were deployed and configured. Apache Atlas was installed initially, followed by the subsequent installation of Apache Ranger and Kafka. The computer storage served as the designated data hub for streamlined data processing.

Stepwise Procedure:
1. Apache Atlas Installation: Apache Atlas was installed on the VM to facilitate metadata management (Figure 15).

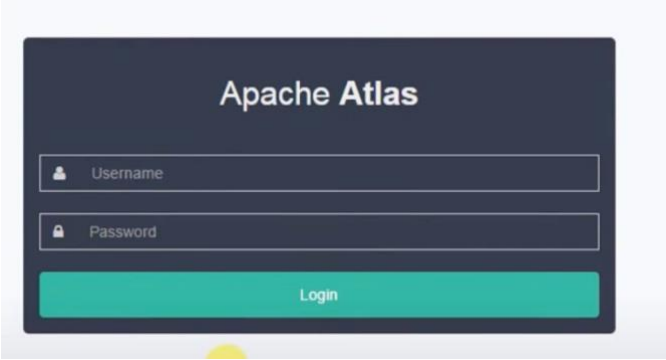


Figure 15. Apache Atlas Login Page

2. Apache Ranger Installation: Apache Ranger was installed and configured in tandem with Apache Atlas for testing the developed security model.
3. Kafka Integration: Kafka was integrated into the system to optimize data processing and distribution (Figure 16).

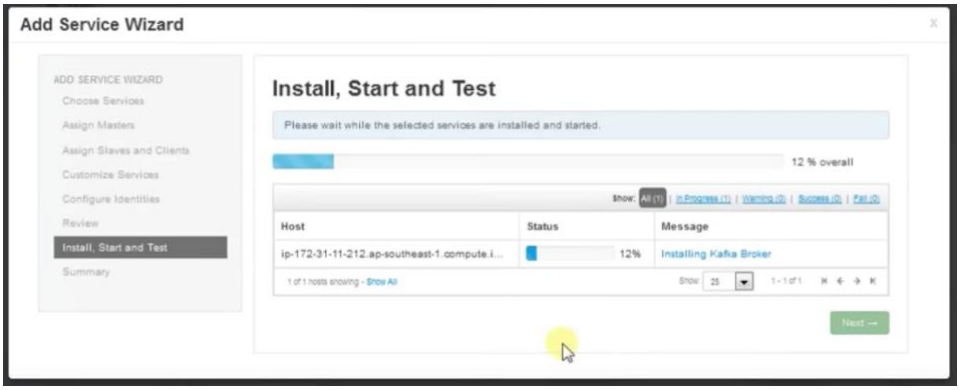


Figure 16. Kafka Integration with Apache Atlas

4. Data-Hub Configuration: The computer storage was designated as the central data hub for efficient data organization.

4.8.2.3 Testing and Validation

This crucial phase involved thorough testing and validation of the data-centric model to ensure its robustness and effectiveness.

Stepwise Procedure:
1. Security Model Testing: Apache Ranger and Apache Atlas were rigorously tested to validate the efficacy of the developed security model.

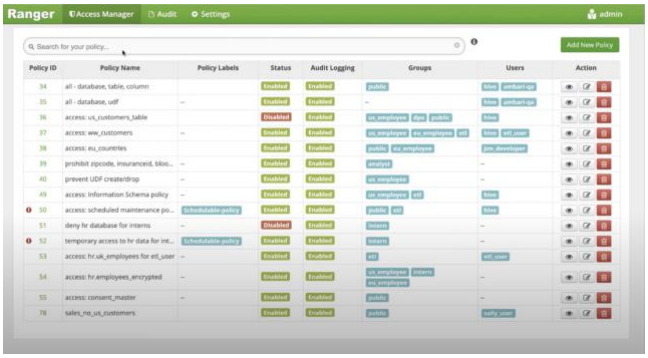


Figure 17. Apache Ranger configuration Page

2. Kafka Integration Testing: The integration of Kafka was tested to ensure seamless data processing.

4.8.2.4 Data Flow and Connectivity Testing

A critical aspect of the proof of concept involved assessing the flow of data within the architecture and ensuring seamless connectivity between components.

Stepwise Procedure:
1. Data Flow Analysis: The flow of data within the architecture was analyzed to identify potential bottlenecks.
2. Connectivity Testing: The connectivity between Apache Atlas, Apache Ranger, Kafka, and the data hub was tested for optimal performance (Figure 18).

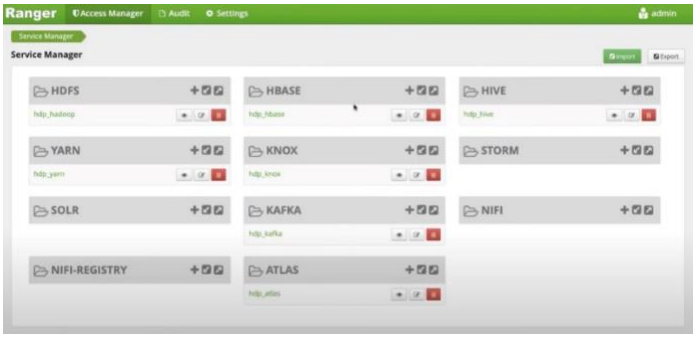


Figure 18. Apache Atlas, Apache Ranger and Kafka Configuration

4.8.2.5 Visualization and Analysis Setup

To enhance data visualization and analysis, Elasticsearch and Kibana were integrated into the architecture using Logstash and Docker.

Stepwise Procedure:
1. Elasticsearch and Kibana Integration: Elasticsearch and Kibana were configured to complement Kafka for streamlined data visualization (Figure 19).

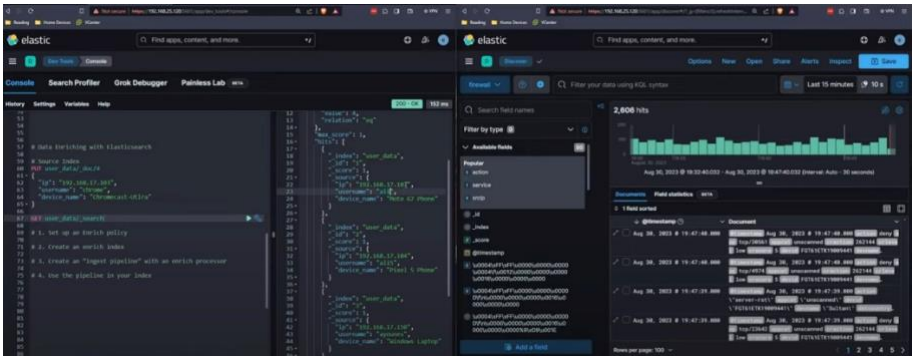


Figure 20. Data Integration to Elastic Search using Kafka.

2. Logstash and Docker Integration: Logstash and Docker were employed to facilitate the efficient flow of data for visualization and analysis (Figure 20 and Figure 21).

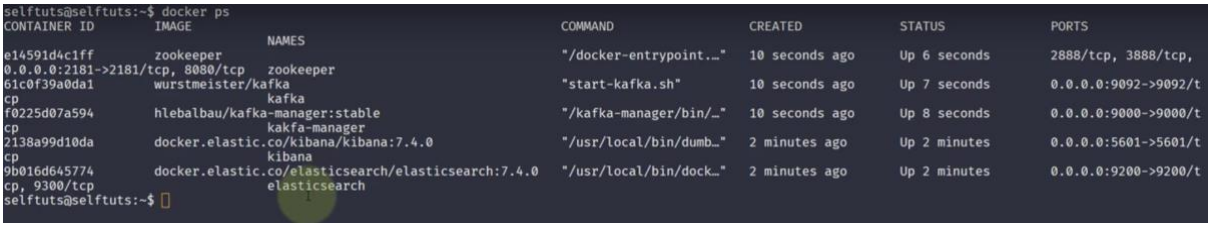


Figure 21. Kafka integration with Kibana and Elasticsearch

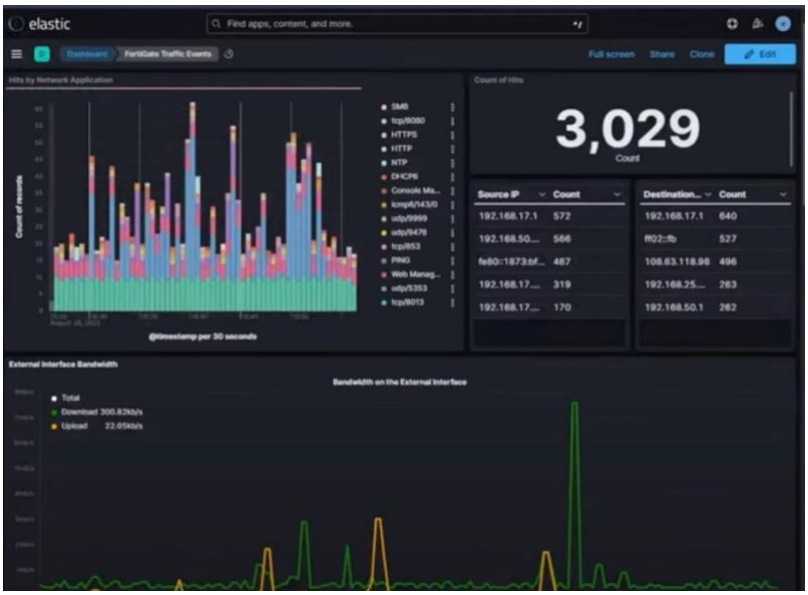


Figure 22. Analytics Dashboard

These stepwise procedures and corresponding diagrams provide a comprehensive view of the proof of concept conducted in the virtualized lab environment, offering clarity on the deployment, testing, and validation processes within the data-centric architecture.

4.8.2 Application of the Secure Grade Distribution Scheme

The feasibility and practicality of the proposed scheme for enhancing security and privacy in educational environments, particularly within the context of Moodle integration, were assessed through a proof of concept (Appendix B). This section outlines the steps involved in developing, implementing, and evaluating the scheme within a controlled lab environment. The proof of concept, however, does not employ HSM; instead, all cryptographic keys are kept in a file inside Moodle.

4.8.2.1 Lab Environment Setup

A virtualized lab environment was created, consisting of three distinct Virtual Machines (VMs) to represent instructors, staff, and students. These VMs were configured to operate on the same network, allowing for seamless communication.

4.8.2.2 Moodle Installation and Configuration

Moodle, the widely used open-source Learning Management System, was installed on each of the VMs, mimicking a real educational environment. The installation and configuration encompassed the web server setup, database creation, and Moodle initialization. A shared folder was established on the instructor's VM for resource sharing.

4.8.2.3 Scheme Plugin Development

A custom scheme plugin was developed to implement secure grade distribution features within Moodle. The plugin integrated Advanced Encryption Standard (AES) encryption, Diffie-Hellman key exchange, and Message Integrity Code (MIC) verification. The development environment included PHP tools and a code editor.

4.8.2.4 Plugin Installation and Activation

The developed scheme plugin was uploaded and activated on each of the Moodle instances representing instructors, staff, and students. This enabled the secure grade distribution features across the roles.

4.8.3 Testing and Validation

A series of tests were conducted to verify the functionality of the scheme plugin:

4.8.3.1 Key Exchange

The Diffie-Hellman key exchange was initiated between the instructor and each student to establish a shared secret key (Figure 23).

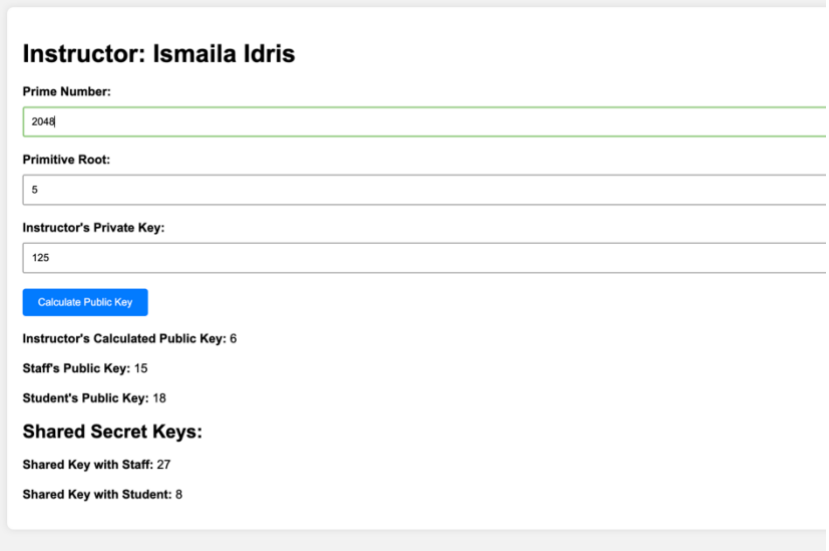


Figure 23. Key Exchange using Diffie-hellman

4.8.3.2 AES Encryption and Decryption

Instructors encrypted grades and shared them with staff and students. Recipients successfully decrypted the ciphertext using the shared secret key (Figure 22).

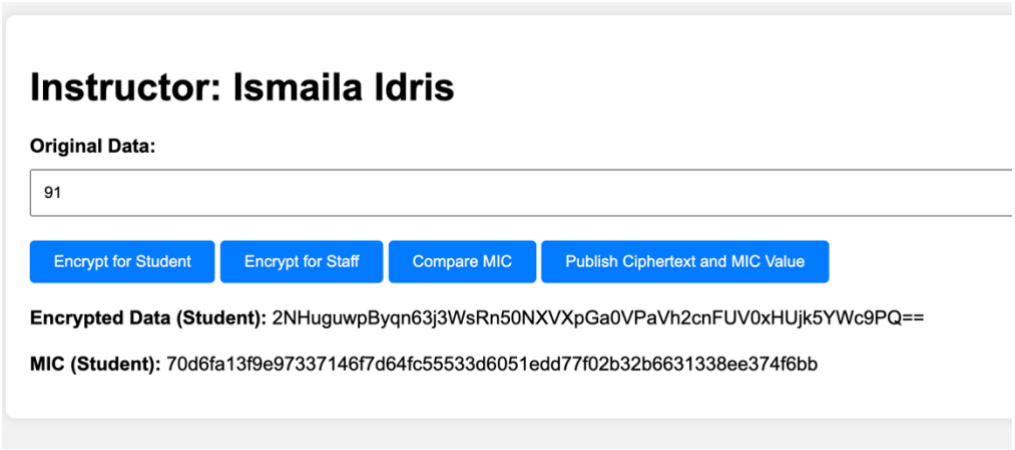


Figure 23. AES Encryption and MIC generation

4.8.3.3 MIC Verification

Recipients verified the integrity of the received grades using Message Integrity Code (MIC) validation (Figure 24).

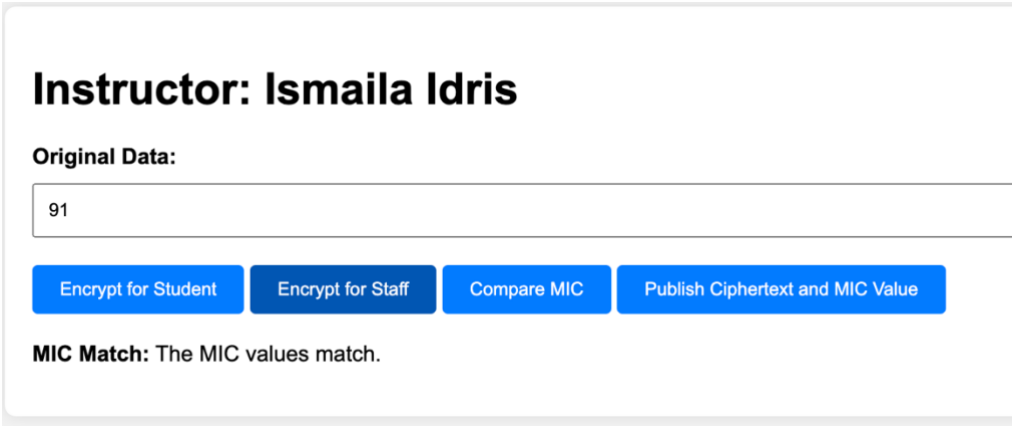


Figure 24. MIC Verification

4.8.3.4 Evaluation and Feedback

Feedback was gathered from participants who represented the roles of instructors, staff, and students in the lab environment. The feedback aimed to evaluate the ease of use, security, and effectiveness of the scheme.

CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary

This research embarked on a comprehensive exploration of data-centric architecture tailored to the context of digital learning universities, particularly within the West African region. The study initiated with a meticulous survey methodology designed to capture insights from technical staff actively engaged with diverse university systems and portals. Employing a combination of survey queries and case studies, the research undertook a robust comparative analysis. The data collection process was methodically crafted, involving the development of survey questions, rigorous validity testing, wide-scale dissemination through established channels, and subsequent analysis employing both a four-point Likert scale and the statistical tool SPSS.

The participant pool, consisting primarily of technical staff, ensured the generation of focused and pertinent data. The study impressively secured responses from 93 universities, surpassing the initially set target of 70%, thereby demonstrating a robust and representative sample. The research design effectively addressed key research questions, encompassing critical aspects such as data architectures, cyber threats, countermeasures, and the role of data in decision-making within university environments. This comprehensive approach to data collection and analysis facilitated the extraction of valuable insights and trends.

Moving forward, the study undertook a meticulous comparison of data architectures, unveiling specific criteria for evaluation. This involved scrutinizing methodologies, understanding data sources, and appraising various storage solutions. The research identified and classified 109 e-learning solution use cases, offering a detailed breakdown based on the data architectures they employed. The findings from this analysis critically underscored the limitations associated with data-driven architecture in terms of security, leading to a compelling recommendation for the adoption of data-centric architecture within e-learning environments.

Continuing the exploration, the study delved into the intricate realm of access control models, providing a nuanced comparison and evaluation. A particular emphasis was placed on the selection of a suitable model for digital learning universities, with Role-Based Access Control

(RBAC) emerging as a pragmatic choice for simplifying access control administration and optimizing performance within the educational context.

The research then navigated through the design and conceptualization of a data-centric model specifically tailored to the unique demands of digital learning universities. This involved a systematic consideration of components such as data sources, connectors, data hubs, governance and security frameworks, data as service, user-centric apps, analytic factories, and analytics as a service. Each component was intricately examined to ensure seamless integration and efficiency within the designed model.

Security stood as a paramount concern throughout the study, prompting a detailed exploration of security models and the subsequent design of a robust framework. The risk assessment meticulously scrutinized various data categories, including personal identifiable information, research data, financial data, academic records, library data, administrative records, and data from digital devices. The comprehensive evaluation led to the incorporation of a security model bolstered by Apache Ranger and Apache Atlas, aimed at fortifying data governance and security within the digital learning environment.

To validate the proposed data-centric architecture and security model, a proof of concept was meticulously executed. This involved the identification of the most suitable storage solutions, particularly focusing on the unique demands of digital learning universities. The study advocated for a hybrid storage approach, leveraging both local and cloud storage, with a crucial emphasis on encrypting all data in local storage using the AES encryption standard. This multifaceted approach aimed to optimize performance, ensure cost efficiency, and enhance data availability and resilience.

In the final stages of the research, the study strategically introduced Kafka, a high-performance data streaming platform, to enhance the overall architecture. The inclusion of Kafka underscored the commitment to leveraging cutting-edge technologies for seamless data processing and distribution within the digital learning landscape.

In summation, this research unfolds as a comprehensive and meticulously executed endeavor, navigating through the intricacies of data-centric architecture in digital learning universities. From the nuanced survey methodology to the design of a tailored data-centric model, the study

contributes valuable insights, recommendations, and a concrete proof of concept to the evolving landscape of educational technology and data management within the West African context.

5.2 Conclusion

In conclusion, this research has traversed the intricate landscape of data-centric architecture in the context of digital learning universities, with a specific focus on the unique demands and challenges within the West African region. The study was initiated with a robust survey methodology, engaging technical staff from numerous universities to garner rich insights and perspectives. The overwhelming response from 93 universities not only exceeded the set target but also ensured a comprehensive and representative dataset for analysis.

The comparative analysis of data architectures shed light on the limitations of data-driven architecture within the e-learning domain, paving the way for a compelling recommendation in favour of data-centric architecture. The research identified and classified 109 e-learning solution use cases, offering a detailed breakdown based on the data architectures they employed. This classification not only contributes to the academic understanding of prevailing trends but also provides practical insights for decision-makers in the educational technology landscape.

A critical aspect of the research focused on the meticulous design of a data-centric model tailored to the specific needs of digital learning universities. Components such as data sources, connectors, data hubs, governance and security frameworks, data as service, user-centric apps, analytic factories, and analytics as a service were intricately examined, ensuring a holistic and seamlessly integrated architecture.

Security remained a paramount concern throughout the study, prompting an in-depth exploration of security models and the subsequent design of a robust framework. The risk assessment, covering various data categories, facilitated the development of a security model fortified by Apache Ranger and Apache Atlas. This model, designed to enhance data governance and security, stands as a pivotal contribution to the evolving field of cybersecurity within educational institutions.

The validation of the proposed data-centric architecture and security model through a proof of concept further strengthens the practical relevance of the research. The advocacy for a hybrid storage approach, coupled with encryption standards for local storage, reflects a nuanced understanding of the balance required between performance, cost efficiency, and data resilience within digital learning environments.

The strategic introduction of Kafka, a high-performance data streaming platform, adds a layer of sophistication to the overall architecture, emphasizing the research's commitment to leveraging cutting-edge technologies for seamless data processing and distribution.

In essence, this research not only advances academic knowledge in the realm of data-centric architecture but also offers actionable insights and recommendations for digital learning universities in West Africa. The comprehensive and meticulously executed nature of this study positions it as a valuable contribution to the evolving landscape of educational technology and data management, with implications for policymakers, educators, and technology practitioners in the region.

5.3 Recommendations

Based on the comprehensive findings and insights derived from this research, several recommendations are put forth to guide digital learning universities, policymakers, and technology practitioners in enhancing their data-centric architecture and cybersecurity frameworks:

1. Adoption of Data-Centric Architecture: Digital learning universities in West Africa are encouraged to transition towards a data-centric architecture. This shift should be underpinned by a thorough assessment of the specific needs and challenges within each institution. Embracing a data-centric model will contribute to improved scalability, flexibility, and efficiency in managing the vast amounts of data generated in the e-learning ecosystem.
2. Integration of Hybrid Storage Solutions: Considering the diverse data types and storage requirements in digital learning environments, the adoption of hybrid storage solutions is recommended. This approach allows for a balanced and cost-effective allocation of storage resources, leveraging the strengths of both local and cloud

storage. Implementation should also prioritize robust encryption standards for local storage to enhance data security.

3. Implementation of Apache Ranger and Apache Atlas: To fortify data governance and security, universities are advised to implement robust frameworks such as Apache Ranger and Apache Atlas. These tools provide a comprehensive set of features for access control, policy enforcement, and metadata management. Customization based on specific institutional requirements is recommended to ensure a tailored and effective security model.
4. Leveraging Kafka for Streamlining Data Processing: The incorporation of Kafka as a high-performance data streaming platform is recommended to streamline data processing and distribution. This technology can enhance the real-time capabilities of digital learning platforms, facilitating efficient data flow and analysis. Universities should consider the scalability and adaptability of Kafka to meet the evolving demands of e-learning.
5. Continuous Security Training and Awareness: Recognizing the dynamic nature of cybersecurity threats, universities should invest in continuous training and awareness programs for staff and students. This proactive approach will foster a culture of cybersecurity awareness and responsible data handling, reducing the risk of security breaches.
6. Regular Updates and Audits: It is imperative for institutions to prioritize regular updates of software, security protocols, and data management policies. Conducting periodic security audits and assessments will help identify vulnerabilities and ensure that the implemented data-centric architecture remains resilient against emerging threats.
7. Collaboration and Information Sharing: Digital learning universities are encouraged to foster collaboration and information sharing within the academic community. Establishing forums or consortia where institutions can share insights, best practices, and challenges related to data-centric architecture and cybersecurity will contribute to collective resilience.
8. Research and Development Initiatives: Investing in research and development initiatives specific to data-centric architecture and cybersecurity in the context of West African digital learning environments is recommended. This proactive stance will contribute to the evolution of tailored solutions that address regional nuances and challenges.

By embracing these recommendations, digital learning universities can not only strengthen their data-centric architecture but also enhance the overall cybersecurity posture, ensuring a secure and efficient learning environment for students and faculty.

5.4 Contributions to Knowledge

This research makes significant contributions to the field of data-centric architecture and cybersecurity within the context of digital learning universities, particularly in West Africa. The key contributions include:

a. Comprehensive Survey and Comparison: The research provides a thorough survey and comparison of data architectures employed in digital learning universities, identifying 109 e-learning solution use cases. This comprehensive analysis sheds light on the diversity of data architectures, paving the way for informed decision-making in selecting the most suitable model.
b. Tailored Data-Centric Model Design: A data-centric architecture model specifically designed for digital learning universities in West Africa is proposed. The model considers the unique challenges and requirements of the region, offering a practical blueprint for institutions aiming to enhance their data management capabilities.
c. Robust Security Model: The research contributes a detailed design of a security model tailored to the data-centric architecture in digital learning universities. This includes the integration of Apache Ranger and Apache Atlas for data governance and security, providing a robust framework to safeguard sensitive information.
d. Proof of Concept: A practical proof of concept is presented, demonstrating the viability and effectiveness of the proposed data-centric architecture and security model. The implementation of Kafka for streamlined data processing adds a practical dimension to the research, showcasing real-world applicability.
e. Insights into Storage Solutions: The research offers insights into storage solutions tailored to the needs of digital learning universities. The discussion on hybrid storage, encryption practices, and the selection of suitable storage options provides valuable guidance for institutions grappling with the management of vast and diverse datasets.

f. Recommendations for Cybersecurity Practices: The research contributes actionable recommendations for enhancing cybersecurity practices in digital learning universities. The emphasis on continuous security training, regular updates, and collaboration underscores the proactive measures needed to mitigate evolving cybersecurity threats.

g. Contextualization for West Africa: The research contextualizes its findings and recommendations within the specific socio-economic and technological landscape of West Africa. This regional focus ensures that the proposed solutions align with the unique challenges faced by digital learning institutions in this geographical context.

Overall, these contributions advance the understanding and implementation of data-centric architecture and cybersecurity practices in digital learning universities, with implications extending beyond the specific region to benefit educational institutions globally.

5.5 Future Research Directions

The research conducted in this study opens avenues for future investigations in several key areas related to data-centric architecture and cybersecurity in digital learning universities. Some potential directions for future research include:

1. Advanced Security Measures: Future research could delve deeper into innovative security measures and technologies that can further fortify data-centric architectures in digital learning universities. Exploring emerging encryption techniques, biometric authentication, and anomaly detection systems could enhance the resilience of cybersecurity frameworks.
2. Adaptability to Emerging Technologies: As technology continues to evolve, future research can explore the adaptability of data-centric architectures to emerging technologies such as blockchain, artificial intelligence, and edge computing. Investigating how these technologies can be integrated to enhance data security and processing efficiency would be valuable.
3. Global Comparative Studies: Conducting comparative studies on data-centric architectures and cybersecurity practices across various regions and continents could provide insights into the contextual differences and common challenges faced by

digital learning universities globally. This comparative approach can inform best practices that are adaptable to diverse environments.

4. User-Centric Design: Future research can focus on refining user-centric applications within data-centric architectures. Exploring ways to improve the user experience, accessibility, and personalization of educational platforms can contribute to more effective learning environments.
5. Longitudinal Studies on Security Efficacy: Conducting longitudinal studies to assess the long-term efficacy of implemented security measures would be beneficial. Tracking the performance of security models over time can reveal potential vulnerabilities and inform the development of adaptive cybersecurity strategies.
6. Integration of Ethical Considerations: Future research can explore the ethical implications of data-centric architectures in digital learning universities. Investigating issues such as data privacy, consent, and responsible use of educational data can contribute to the development of ethically sound frameworks.
7. Scalability and Resource Optimization: As digital learning platforms continue to grow, scalability and resource optimization become critical. Future research can focus on developing strategies for optimizing data-centric architectures to handle increasing volumes of data while ensuring efficient resource utilization.
8. Collaboration and Information Sharing: Exploring collaborative approaches and information-sharing mechanisms among digital learning universities can enhance collective cybersecurity efforts. Research in this area could investigate models of collaboration, threat intelligence sharing, and joint response strategies.

By pursuing these future research directions, scholars and practitioners can contribute to the ongoing evolution of data-centric architectures and cybersecurity practices in digital learning universities, fostering a more secure and adaptable educational landscape.

REFERENCES:

1. Abdelsalam, M., Idrees, A. M., & Shokry, M. (2023). A Proposed Model for Improving the Reliability of Online Exam Results Using Blockchain. *IEEE Access*. https://ieeexplore.ieee.org/abstract/document/10216975/
2. Aborode, A., Anifowoshe, O., Ayodele, T. I., Iretiayo, A. R., & David, O. O. (2020). *Impact of COVID-19 on education in sub-Saharan Africa*. https://www.preprints.org/manuscript/202007.0027
3. Adnan, M. (2020). Online learning amid the COVID-19 pandemic: Students perspectives. *Journal of Pedagogical Sociology and Psychology*, *1*(2), 45–51. https://doi.org/10.33902/JPSP.2020261309
4. Adobe. (2021). *Adobe Privacy Centre*. https://www.adobe.com/africa/privacy/policy.html
5. Aissaoui, K., & Azizi, M. (2017). El-security: E-learning systems security checker plug-in. *Proceedings of the 2nd International Conference on Big Data, Cloud and Applications*, 1–6.
6. Alassery, H. A. A. F. (2021). Securing fog computing for e-learning system using integration of two encryption algorithms. *Journal of Cybersecurity*, *3*(3), 149.
7. Alfonso, F. (2018, February 13). Data-driven versus data-centric. *Stratio*. https://blog.stratio.com/datadriven-versus-datacentric/
8. Allen, I. E., & Seaman, J. (2017). Digital Compass Learning: Distance Education Enrollment Report 2017. *Babson Survey Research Group*. https://eric.ed.gov/?id=ed580868
9. Al-Malah, D. K. A.-R., Aljazaery, I. A., Alrikabi, H. T. S., & Mutar, H. A. (2021). Cloud computing and its impact on online education. *IOP Conference Series: Materials Science and Engineering*, *1094*(1), 012024. https://iopscience.iop.org/article/10.1088/1757-899X/1094/1/012024/meta
10. Al-Naser, A., Rasheed, M., Irving, D., & Brooke, J. (2013). *A Data-Centric Approach to Data Provenance in Seismic Imaging Data*. cp. https://doi.org/10.3997/2214-4609.20130200
11. Al-Sherideh, A. S., Maabreh, K., Maabreh, M., Al Mousa, M. R., & Asassfeh, M. (2023). Assessing the Impact and Effectiveness of Cybersecurity Measures in e-Learning on Students and Educators: A Case Study. *International Journal of Advanced Computer Science and Applications*, *14*(5). https://search.proquest.com/openview/175afa53a183a601946e5c20a9abae52/1?pq-origsite=gscholar&cbl=5444811
12. Arabi, A. A. M. (2021). A zero-trust model-based framework for managing of academic dishonesty in institutes of higher learning. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(6), 5381–5389.
13. Ascend. (2020). *Data-Informed, Data-Driven, and Data-Centric: What's the Difference?* Ascend Venture Capital. https://www.ascendstl.com/press/2020/4/28/data-driven-and-data-centric-whats-the-difference
14. Aulakh, K., Roul, R. K., & Kaushal, M. (2023). E-learning enhancement through educational data mining with Covid-19 outbreak period in backdrop: A review. *International Journal of Educational Development*, *101*, 102814. https://doi.org/10.1016/j.ijedudev.2023.102814
15. Bates, B. (2019). Learning theories simplified: And how to apply them to teaching. *Learning Theories Simplified*, 1–384.

16. Bellaj, S. (2021). *Etakwin*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/etakwin
17. Bervell, B., & Umar, I. N. (2017). A Decade of LMS Acceptance and Adoption Research in Sub-Sahara African Higher Education: A Systematic Review of Models, Methodologies, Milestones and Main Challenges. *EURASIA Journal of Mathematics, Science and Technology Education*, *13*(11). https://doi.org/10.12973/ejmste/79444
18. Bezovski, Z., & Poorani, S. (2016). The evolution of e-learning and new trends. *Information and Knowledge Management*, *6*(3), 50–57.
19. Bhatia, M., & Maitra, J. K. (2018). E-learning Platforms Security Issues and Vulnerability Analysis. *2018 International Conference on Computational and Characterization Techniques in Engineering & Sciences (CCTES)*, 276–285. https://doi.org/10.1109/CCTES.2018.8674115
20. Bill, B. (2021). *Blackboard Inc. [NASDAQ:BBBB]: The Digital Learning Champions*. CIOReview. https://education.cioreview.com/vendor/2017/blackboard_inc._[nasdaq:bbbb]
21. Boh Podgornik, B., Dolničar, D., Šorgo, A., & Bartol, T. (2016). Development, testing, and validation of an information literacy test (ILT) for higher education. *Journal of the Association for Information Science and Technology*, *67*(10), 2420–2436.
22. Brown, C. (2021). *Coassemble*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/coassemble
23. Butler, J. (2021). *Nimble LMS*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/nimble-lms
24. Campus. (2021). *Award Winning Cloud ERP System for University, College, Cloud ERP for Small Companies*. https://campuslabs.in/campus-erp/
25. Carol, D. (2021, February 4). *The Difference Between Data-centric and Data-driven*. Applied Software. https://www.asti.com/the-difference-between-data-centric-and-data-driven/
26. Carvalho Ota, F. K., Augusto Meira, J., Frank, R., & State, R. (2020). Towards Privacy Preserving Data Centric Super App. *2020 Mediterranean Communication and Computer Networking Conference (MedComNet)*, 1–4. https://doi.org/10.1109/MedComNet49392.2020.9191550
27. CIO, Re. (2021). *SharedBook: You Can Have It Both Ways*. CIOReview. https://education.cioreview.com/vendor/2017/sharedbook
28. claned. (2021). *Privacy policy*. Claned. https://claned.com/privacy-policy/
29. Crompton, H. (2013). The benefits and challenges of mobile learning. *Learning and Leading with Technology*, *41*. https://digitalcommons.odu.edu/cgi/viewcontent.cgi?article=1136&context=teachinglearning_fac_pubs
30. Cyoy, R. B. (2022). *Framework for Effective Management of Cyber Security on E-learning Platforms in Public Universities in Kenya* [PhD Thesis, university of nairobi]. http://erepository.uonbi.ac.ke/handle/11295/161726
31. Dabbagh, N., & Kitsantas, A. (2012). Personal Learning Environments, social media, and self-regulated learning: A natural formula for connecting formal and informal learning. *The Internet and Higher Education*, *15*(1), 3–8.
32. Daniel, J. (2012). Making sense of MOOCs: Musings in a maze of myth, paradox and possibility. *Journal of Interactive Media in Education*, *2012*(3). https://jime.open.ac.uk/article/10.5334/2012-18/
33. Dave, M. (2020). The Data-Centric Revolution: Data-Centric vs. Data-Driven. *TDAN.Com*. https://tdan.com/the-data-centric-revolution-data-centric-vs-data-driven/20288

34. Djeki, E., Dégila, J., & Alhassan, M. H. (2023). E-Learning Challenges and Opportunities in West Africa During COVID-19 Pandemic. *2023 IEEE 12th International Conference on Engineering Education (ICEED)*, 159–164. https://doi.org/10.1109/ICEED59801.2023.10264039
35. Docebo. (2021). *Docebo*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/docebo
36. docebo. (2021). Docebo—Privacy Policy. *Docebo*. https://www.docebo.com/company/privacy-policy/
37. Doust, S. (2021). *Glo™ learn*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/glo
38. Elmaghrabi, A. Y., & Eljack, S. M. (2019). Enhancement of Moodle learning management system regarding quizzes security and stability problems. *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*, 1–7. https://ieeexplore.ieee.org/abstract/document/8769530/
39. Ertmer, P. A., Ottenbreit-Leftwich, A. T., Sadik, O., Sendurur, E., & Sendurur, P. (2012). Teacher beliefs and technology integration practices: A critical relationship. *Computers & Education*, *59*(2), 423–435.
40. Fatokun, F. B., Hamid, S., Norman, A., & Fatokun, J. O. (2019). The impact of age, gender, and educational level on the cybersecurity behaviors of tertiary institution students: An empirical investigation on Malaysian universities. *Journal of Physics: Conference Series*, *1339*(1), 012098.
41. Ferguson, R., Coughlan, T., Egelandsdal, K., Gaved, M., Herodotou, C., Hillaire, G., Jones, D., Jowers, I., Kukulska-Hulme, A., & McAndrew, P. (2019). *Innovating pedagogy 2019: Open university innovation report 7*. https://oro.open.ac.uk/59132/1/innovating-pedagogy-2019.pdf
42. Firat, M., & Bozkurt, A. (2020). Variables affecting online learning readiness in an open and distance learning university. *Educational Media International*, *57*(2), 112–127.
43. GDPR. (2018). *GDPR Compliance | LMS by Mindflash*. https://mindflash.com/gdpr
44. Ghatak, S. R. (2021). *Adobe Captivate Prime*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/adobe-captivate-prime
45. Gogos, R. (2021). *Looop*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/looop
46. Gray, M. (2021). *Xperiencify*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/xperiencify
47. Guo, S., & Zeng, D. (2020). Pedagogical Data Federation toward Education 4.0. *Proceedings of the 2020 The 6th International Conference on Frontiers of Educational Technologies*, 51–55. https://doi.org/10.1145/3404709.3404751
48. gyrus. (2021). *LMS Privacy Policy*. https://www.gyrus.com/privacy-policy
49. Hasan, M. R., Rahman, R., & Zaman, K. (2022). Design an Information Security Framework for University Automation System. *2022 25th International Conference on Computer and Information Technology (ICCIT)*, 454–459. https://ieeexplore.ieee.org/abstract/document/10054997/
50. Hjorth, S.-J. (2021). *CanopyLAB - Social Learning Powered by AI*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/canopylab-social-learning-powered-by-ai
51. Hodges, C. B., Moore, S., Lockee, B. B., Trust, T., & Bond, M. A. (2020). *The difference between emergency remote teaching and online learning*. https://vtechworks.lib.vt.edu/handle/10919/104648
52. Inc, A. L. S. (2021). *Absorb Privacy Policy—LMS Data Security—Absorb LMS Software*. https://www.absorblms.com/support/privacy-policy

53. Ispring. (2021). *ISpring Learn*. https://elearningindustry.com/directory/elearning-software/ispring-learn
54. Jennifer. (2021). *Inquisiq*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/inquisiq
55. Joksimović, S., Kovanović, V., & Dawson, S. (2019). The journey of learning analytics. *HERDSA Review of Higher Education*, *6*, 27–63.
56. Jusas, V., Butkiene, R., Venčkauskas, A., Grigaliūnas, Š., Gudoniene, D., Burbaite, R., & Misnevs, B. (2022). *Sustainable and Security Focused Multimodal Models for Distance Learning. Sustainability 2022, 14, 3414*. s Note: MDPI stays neutral with regard to jurisdictional claims in published …. https://www.academia.edu/download/89627685/pdf.pdf
57. Kale, A. W., Narawade, V. E., & Kothoke, P. M. (2023). 7 A Study on Online Learning Systems' Identification with Security Schemes and Applications. *Online Learning Systems: Methods and Applications with Large-Scale Data*, 73–80.
58. Kampakis, D. S. (2018, October 22). What are the differences between data-driven, data-informed and data-centric? *The Data Scientist*. https://thedatascientist.com/data-driven-data-informed-data-centric/
59. Kapadia, V. (2021). *GyrusAim*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/gyrusaim
60. Kim, H. (2019). Research issues on data centric security and privacy model for intelligent internet of things based healthcare. *ICSES Trans. Comput. Netw. Commun*, *5*, 1–3.
61. Korać, D., Damjanović, B., & Simić, D. (2022a). A model of digital identity for better information security in e-learning systems. *The Journal of Supercomputing*, *78*(3), 3325–3354. https://doi.org/10.1007/s11227-021-03981-4
62. Korać, D., Damjanović, B., & Simić, D. (2022b). A model of digital identity for better information security in e-learning systems. *The Journal of Supercomputing*, *78*(3), 3325–3354. https://doi.org/10.1007/s11227-021-03981-4
63. learn upon. (2021). *LearnUpon LMS - eLearning Industry*. https://elearningindustry.com/directory/elearning-software/learnupon-lms
64. Lewis, N. J., & Orton, P. (2000). The Five Attributes of I Innovative E-Learning. *Training & Development*, *54*(6), 47–47.
65. Li, C., Guo, J., Zhang, G., Wang, Y., Sun, Y., & Bie, R. (2019). A blockchain system for E-learning assessment and certification. *2019 IEEE International Conference on Smart Internet of Things (SmartIoT)*, 212–219.
66. Lu, Y., & Da Xu, L. (2018). Internet of Things (IoT) cybersecurity research: A review of current research topics. *IEEE Internet of Things Journal*, *6*(2), 2103–2115.
67. Lynch, M. (2021). *Absorb LMS*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/absorb-lms
68. Mackey, T. P., & Jacobson, T. E. (2011). Reframing information literacy as a metaliteracy. *College & Research Libraries*, *72*(1), 62–78.
69. Malekos, N. (2021). *LearnWorlds*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/learnworlds
70. Maurice, D. H., Ke, H., Ahmad, F., Wang, Y., Chung, J., & Manganiello, V. C. (2014). Advances in targeting cyclic nucleotide phosphodiesterases. *Nature Reviews Drug Discovery*, *13*(4), 290–314.
71. Means, B., & Neisler, J. (2021). Teaching and learning in the time of COVID: The student perspective. *Online Learning*, *25*(1).
72. Media, S. (2021). *Eurekos LMS*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/eurekos

73. Mihailescu, M. I., Nita, S. L., & Corneliu, P. V. (2020). Applied cryptography in designing e-learning platforms. *The International Scientific Conference ELearning and Software for Education*, *2*, 179–189. https://search.proquest.com/openview/eb907a3aa93f0d1dcf65e585b349dac1/1?pq-origsite=gscholar&cbl=1876338&casa_token=NTS59L-3ZY0AAAAA:Xsps3o2s9q9DnH74w-D338iGOPrIFjxdxyvXtZudCtjEUqwXOeHoMaXbIF5YAXzqB3djw-WHJHw
74. Modesti, P. (2020). Integrating Formal Methods for Security in Software Security Education. *Informatics in Education-An International Journal*, *19*(3), 425–454.
75. Mustofa, M., Ahmadi, R., & Karimullah, I. W. (2020). Islamic Character Education in E-Learning Model: How Should It be Implemented? *Jurnal Sains Sosio Humaniora*, *4*(1), 89–93.
76. Nakagawa, H., Iwasawa, Y., & Matsuo, Y. (2019). Graph-based knowledge tracing: Modeling student proficiency using graph neural network. *IEEE/WIC/ACM International Conference on Web Intelligence*, 156–163. https://dl.acm.org/doi/abs/10.1145/3350546.3352513
77. Papagelis, A. (2021). *TalentLMS*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/talentlms
78. Pappas, G. (2021). *Edsby LMS*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/edsby-lms
79. Pérez, S. O., Díez, C. H., & García, J. A. M. (2017). Applying Security to Moodle Grades. *Proceedings of the International Conference on Security and Management (SAM)*, 117–123. https://www.proquest.com/docview/2139471781/abstract/BB7AE83FF8544693PQ/1
80. Plyer, L., Marcou, G., Perves, C., Schurhammer, R., & Varnek, A. (2022). Implementation of a soft grading system for chemistry in a Moodle plugin. *Journal of Cheminformatics*, *14*(1), 72. https://doi.org/10.1186/s13321-022-00645-0
81. Policy. (2018). *Privacy Policy • DigitalChalk Continuing Education Solutions*. DigitalChalk Continuing Education Solutions. https://digitalchalkeu.wpengine.com/about-digitalchalk/privacy-policy
82. Policy. (2020). *Privacy Policy | LMS by Mindflash*. https://mindflash.com/privacy-policy
83. Ponomarev, D. (2021). *Gurucan*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/gurucan
84. Privacy Policy. (2021). *Inquisiq*. https://inquisiq.com/privacy/
85. pseudonymisation. (2020). *Privacy Policy | Coassemble*. Coassemble ELearning Software. https://coassemble.com/privacy-policy
86. Ramanauskaitė, S., Urbonaitė, N., Grigaliūnas, Š., Preidys, S., Trinkūnas, V., & Venčkauskas, A. (2021). Educational organization's security level estimation model. *Applied Sciences*, *11*(17), 8061.
87. Scott, S. (2021). *Thinkific*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/thinkific
88. Selwyn, N. (2016). Minding our language: Why education and technology is full of bullshit … and what might be done about it. *Learning, Media and Technology*, *41*(3), 437–443. https://doi.org/10.1080/17439884.2015.1012523
89. Setiawan, R., Arif, F. A. S., Putro, J. O., Princes, E., Silalahi, F. T. R., Geraldina, I., Julianti, E., & Safitri, J. (2023). E-Learning Pricing Model Policy for Higher Education. *IEEE Access*, *11*, 38370–38384. https://doi.org/10.1109/ACCESS.2023.3266954
90. Shodeinde, M. (2021). *Claned*. ELearning Industry. https://elearningindustry.com/directory/elearning-software/claned

91. Siemens, G., & Long, P. (2011). Penetrating the fog: Analytics in learning and education. *EDUCAUSE Review*, *46*(5), 30.
92. Sinan, I. I., Degila, J., Nwaocha, V., & Onashoga, S. A. (2022a). Data Architectures' Evolution and Protection. *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 1–6. https://doi.org/10.1109/ICECET55527.2022.9872597
93. Sinan, I. I., Degila, J., Nwaocha, V., & Onashoga, S. A. (2022b). Data Architectures' Evolution and Protection. *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 1–6.
94. Sinan, I. I., Nwoacha, V., Degila, J., & Onashoga, S. A. (2022). A Comparison of Data-Driven and Data-Centric Architectures using E-Learning Solutions. *2022 International Conference Advancement in Data Science, E-Learning and Information Systems (ICADEIS)*, 1–6.
95. Skinner, B. F. (1957). The experimental analysis of behavior. *American Scientist*, *45*(4), 343–371.
96. Sun, N., Ding, M., Jiang, J., Xu, W., Mo, X., Tai, Y., & Zhang, J. (2023). Cyber Threat Intelligence Mining for Proactive Cybersecurity Defense: A Survey and New Perspectives. *IEEE Communications Surveys & Tutorials*. https://ieeexplore.ieee.org/abstract/document/10117505/
97. talent LMS. (2021). *Data Security in LMS - Secure Online Learning System—TalentLMS*. https://www.talentlms.com/security
98. thinkific. (2020). *Security Overview Thinkific Website*. Thinkific. https://www.thinkific.com/security-overview-thinkific-website/
99. Vista. (2021). *Data-centric Architecture—A Different Way of Thinking | Vista Projects*. https://www.vistaprojects.com/blog/data-centric-architecture/
100. Warschauer, M., & Matuchniak, T. (2010). New Technology and Digital Worlds: Analyzing Evidence of Equity in Access, Use, and Outcomes. *Review of Research in Education*, *34*(1), 179–225. https://doi.org/10.3102/0091732X09349791
101. xper. (2020). *Privacy Policy*. https://howto.xperiencify.com/article.php?article=88
102. Zheng, X., Li, Q., & Kong, L. (2010). A Data Storage Architecture Supporting Multi-level Customization for SaaS. *2010 Seventh Web Information Systems and Applications Conference*, 106–109. https://doi.org/10.1109/WISA.2010.18

APPENDIX

7.1 Appendix A

A Survey on Cyber-attacks Faced by Data Architectures in West African Institutions During the COVID-19 Era

Data architecture is a collection of models, policies, rules, and standards used by institutions and organizations to manage which data is collected and how it is kept, processed, and integrated. Each institution has a data architecture that is either data-informed, data-driven, or data-centric.

- Data-informed architecture: Data are collected from many sources, such as external and internal hard drives of computers, flash drives, and so on. A dashboard or excel is used to analyze the data, and the results are used as part of inputs in decision-making.
- Data-driven architecture: In this design, algorithms are used to make decisions based on data collected from various data silos such as the cloud, data lakes, and so on.
- Data-centric architecture: Here, the institution builds a single data model utilized by all information systems in the institution, data science is used as the core in decision making, and all data are integrated and connected using a graph database, removing data redundancy and silos.

ACETEL and ACE-SMIA in conjunction with DSTN, ACE-Partner, IRD, AFD, AAU, and the World Bank are conducting research to provide a secure data architecture that will aid in achieving safer learning environment for west African institutions.

Consequently, this survey was initiated to gain a better understanding of the types data architectures these institutions employ and the types of cyber-attacks/threats they faced during the COVID-19 pandemic; your input will contribute immensely to this research. This survey is strictly for the purpose of research and your responses remain confidential.

1. Please select your gender *
   - Male
   - Female
2. Please select your country*
   - Bénin
   - Burkina Faso
   - Cabo Verde
   - Cote d'Ivoire
   - Gambia
   - Ghana
   - Guinea
   - Guinea-Bissau
   - Liberia
   - Mali

- Mauritania
- Niger
- Nigeria
- Senegal
- Sierra Leone
- Togo
- Other :

3. Please select the type of your institution*
- Public
- Private
- Other:

4.Please select the mode of delivery in your institution*
- Face-to-face
- E-learning
- Blended

5.Please select the age group you belong *
- Below 25 years
- 26 – 35 years
- 36 – 45 years
- 46 – 55 years
- 56 and above

6.Please indicate your current academic level *
- Diploma
- Bachelor degree
- Master's degree
- PhD / Doctorat
- Other:

7.Please indicate, if your institution conducted any training/ workshop during the period of COVID-19 pandemic*
- Yes
- No

8.Please indicate, if your institution conducts any of the following online*
- Application
- Registration
- Lectures
- Examination

9.Please indicate the type of data architecture employ by your institution*
- Data-informed architecture
- Data-driven architecture
- Data-centric architecture

10. To what extent does your institution use the following tools to ANALYSE and REPORT on data you collect and store? *

	Extensively	Moderately	A Little	Not at all
1. Spreadsheets (Charts, counts, pivot tables)	O	O	O	O

2. Website analytics (e.g., Google Analytics).	O	O	O	O
3. Database Database (CRM analytics and reports)	O	O	O	O
4. Specialist tools (e.g., SAS, R, Stata, Python, SPSS, GIS Mapping)	O	O	O	O

11. Which of these best describes your institution's use of data for decision-making?*

- We do not use data at all for decision-making, we rely on gut feeling and experience
- We use data about what happened in the recent past (e.g., last quarter or last year)
- We use past and recent data, including some longer-term trends analysis
- We monitor what's happening now, in real-time, as well as past trends
- We use past, present, and forward-looking data (e.g. forecasting, modelling, and optimization)

12. To what extent has your institution's staff use data and analysis to influence or inform their activities and decisions in the following areas?*

	Extensively	Moderately	A Little	Not at all
• Students' satisfaction with their teaching & learning experience	O	O	O	O
• Need for student and/or staff engagement	O	O	O	O
• Academic development and performance review	O	O	O	O
• Research opportunities and potential research partners	O	O	O	O
• Environmental impacts from institutional activities	O	O	O	O
• Other societal impacts from institutional activities	O	O	O	O
• Mid and long-term strategic planning	O	O	O	O

13. Which of these best describes how your institution is planning for improvement in data?

- There is no plan and no intention to make one
- There is no plan but we are thinking we should have one
- We are actively creating a plan
- We have a plan and are implementing it
- There is a regular cycle of planning, implementation and review

14.Please indicate if your institution is a victim of cyber-attacks*

- Yes
- No

15. If Yes in the above, please indicate; the type of cyber-attack your institution faced SQL Injection

- Denial of service (DOS)
- Ransomware
- Virus
- Worm
- Phishing
- Other:

16.Please indicate; the data protection technique(s) employed by your institutions*

- Firewall
- Anti-virus
- Intrusion detection system
- Intrusion prevention system
- Other:

17.Are you satisfied with your institution data protection technique(s) / Êtes-vous satisfait de la ou des techniques de protection des données de votre établissement ?

- Yes / Oui
- Neutral / neutre
- No / Non

18.How would you rate the level of your institution cyber countermeasures / Comment évaluez-vous le niveau des contre-mesures informatiques de votre institution*

- Poor
- Fair
- Good
- Excellent
- Don’t Know

19. How often do you attend workshop/ training in cybersecurity?*

- Not at all / Pas du tout
- Every 3 months
- Every 6 months
- Every 12 months
- Other:

20.How would you rate the level of your cybersecurity knowledge and skill? *

- Poor
- Fair
- Good
- Excellent
- Don’t Know

Thank you

Back

Submit

Clear form

7.2 Appendix B

Secure grade distribution code is available at: https://github.com/iisinan/grade-distribution-scheme/tree/main/aes_encryption 2

COVER PAGE

AN ENHANCED THREAT CLASSIFICATION FRAMEWORK FOR ELECTRONIC HEALTH SYSTEMS IN NIGERIA

EDITH ABENGOWE

B.Sc. (HONS.) (NOUN), M.Sc. (NOUN)

ACE22250014

AUGUST, 2024

TITLE PAGE

AFRICA CENTRE OF EXCELLENCE FOR TECHNOLOGY ENHANCED LEARNING (ACETEL)

AN ENHANCED THREAT CLASSIFICATION FRAMEWORK FOR ELECTRONIC HEALTH SYSTEMS IN NIGERIA

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY IN

CYBERSECURITY

BY

EDITH ABENGOWE

ACE22250014

SUPERVISORS:

1. **Dr. Uyinomen Ekong**
2. **Dr. Saheed Kayode**
3. **Mr. Felix Rishamma**

SEPTEMBER, 2024

DECLARATION

I, Edith Abengowe, the author of this dissertation entitled "An Enhanced Threat Classification Framework for Electronic Health Systems in Nigeria", affirm that it is my original work and that it has not been submitted, in whole or in part, in any previous application for a degree and I will not present it, or cause it to be presented, for a degree in another institution. All sources of information have been appropriately acknowledged using references and other acceptable methods.

NAME: EDITH ABENGOWE

SIGNATURE: _____

DATE: _____

CERTIFICATION

This is to certify that this dissertation: "An Enhanced Threat Classification Framework for Electronic Health Systems in Nigeria" submitted by Edith Abengowe (ACE22250014), in partial fulfillment of the requirements for the award of the degree of Doctor of Philosophy in Cybersecurity, meets the regulations governing the award of PhD degree of the Africa Centre of Excellence and Technology Enhanced Learning (ACETEL). The work has made original contribution to knowledge. It has been examined and approved by the following Supervisors:

(a) Supervisor

Signature……. Date: ……

Name:…….. Rank:……

(b) Co-Supervisor

Signature……. Date: ……

Name:…….. Rank:……

(c) Co-Supervisor

Signature……. Date: ……

Name:…….. Rank:……

DEDICATION

This dissertation is dedicated to my spouse, Emmanuel Abengowe for believing in me and standing by me throughout the writing of this dissertation and to my children, Barbara and Pearl, for their support and understanding.

ACKNOWLEDGEMENTS

My profound gratitude goes to Almighty God for His faithfulness and grace, and for granting me the wisdom, strength, and perseverance to complete this research.

I would like to express my deepest gratitude to Dr. Uyinomen Ekong, my Supervisor, for her guidance, support, and invaluable feedback throughout this research. I am also grateful to Mr. Felix Rishamma, my Co-Supervisior, for his insightful comments and suggestions.

My sincere appreciation also goes to the staff of ACETEL for their academic and administrative support during my study. I would also like to thank my HOD, Dr. Adeyinka Abiodun for her valuable comments and support that greatly enhanced the research process. Each time I call, she responds.

My heartfelt thanks go to my family and friends, especially Aisha Adamu, Charity Nuhu (my HOU), Aisha Ali, Faith Azemobor, and Esther Lasisi for their unwavering support and encouragement. I would not fail to mention my amazing boss (Dr. Olayemi Nwankwo) for being my backbone during the course of this research.

To everyone who has supported me along this journey, thank you. This achievement would not have been possible without you.

Finally, I extend my appreciation to all those who assisted in getting materials for this study, particularly Dr. Oyong, you came through for me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ACL	Access Mitigate List
AES	Advanced Encryption Standard
AP	Access Point
AP	Aptoide
API	Application Programming Interface
APK	Android Application Package
ARP	Address Resolution Protocol
ART	Android Run Time
BLE	Bluetooth Low Energy Protocol
BMP	Bit-map Image Format
BSS	Basic Service Set
CRC	Cyclic Redundancy Check
DCT	Discrete Cosine Transformation
DDK	Knowledge Discovery and Datamining
DES	Data Encryption Standard
DEX	Delvik Executable Format
DFT	Discrete Fourier Transform Technique
D-H	Diffie-Helman algorithm
DPI	Deep Packet Inspection
DS	Distributed system
DWT	Discrete Wavelet Transformation
EAP	Extensible Authentication Protocol
ESS	Extended service set
FIPS	Federal Information Processing Standard

FTP	File Transfer Protocol
HAS	Human Auditory System
HMAC	Hash Message Authentication Code
HVS	Human Visual System
IANA	Internet Assigned Network Authority
IBSS	Independent basic service set
ICMP	Internet Mitigate Message Protocol
ICV	Integrity Checksum value
IdP	Identity Provider
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronic Engineering
IoT	Insecurity of Things
IoT	Internet of Things
IP	Internet Protocol
IV	Initialization vector
JPEG	Joint Photographic Experts Group
LSB	Least Significant Bits
MAC	Media Access Control
MD2	Message Digest 2
MIC	Message Integrity Check
MITM	Man-In-The-Middle attack
MPD	Multi-pixel Differencing
MPEG	Moving Picture Experts Group
MSE	Mean Square Error
NIC	Network Interface Card

NLP	Natural Language Processing
OTN	Optical Transport Network
OTP	One Time Password
Ping	Packet Internetwork reachability to the target
PRNG	Pseudorandom Number Generator
PSNR	Peak Signal to Noise Ratio
PSTN	Public Switched Telephone Network
PVD	Pixel Value Differencing
RFP	Request for proposal
RGB	Red, Green, and Blue values
RMSE	Root Mean Square Error
RSA	Rivest-Shamir-Adleman algorithm
RTU	Remote Terminal Unit
SaaS	Software as a Service
SAML	Security Assertion Markup Language
SAT	Site Acceptance Test
SHA	Secure Hash Algorithm
SMS	Short Message Services
SNMP	Simple Network Monitoring Protocol
SSID	Service Set Identifier
SSIM	Structural Similarity Index
SSL	Secure Socket Layer
SSO	Single sign on
STK	Short Term Key
TDR	Time Domain Reflectometer

TK Temporary Key

TKIP Temporary key integrity protocol

VPN Virtual Private Network

VS Virus Share

VSM Vector Space Model

WAP Wireless Access Protocol

WEP Wired Equivalent Privacy protocol

Wi-Fi Wireless Fidelity

WPA Wi-Fi Protected Access

WS Web service

WSN Wireless Sensor Network

XACML Extensible Access Control Markup Language

XML Extensible Markup Language

LIST OF APPENDICES

PUBLISHED WORK

ABSTRACT

The security of healthcare systems and patients' records privacy are constantly threatened by malware through attacks on them and expose them to ridicule, embarrassment and possible litigation on hospital management for breach of ethics on the handling of patients' records. Attack agents include viruses, worms, Trojans such as ransomware, key loggers and rootkits; while attack types include denial of service (DOS), probe, remote to local (R2L) and user to root (U2R). To curb the menace of malware threats, this research work developed a Framework using machine learning (ML) tools, combined them using ensemble learning technique such as bagging. The dataset used is Network Security laboratory – knowledge discovery in databases (NSL-KDD), obtained from Kaggle, a public data repository. It is categorized into normal class and attack types. This dataset is imbalanced with normal features being much more than attack types, especially PROBE, R2L and U2R types. For effective performance of the trained models, the dataset had to be balanced using Synthetic minority oversampling technique (SMOTE) and extracted using Pearson correlation and information gain, then normalized and selected using principal component analysis (PCA). The preprocessed data is then split into training dataset (80%) with 125,973 records and test dataset (20%) with 22,544 records using train_test_split() function. ML tools used include Random Forest algorithm that trains base classifiers such as k_Nearest_Neighbors (KNN), Naïve Bayes (NB), Decision tree (DT), and Logistic regression (LR). The predictions of trained classifiers are aggregated using majority voting scheme into soft vote classifier, which is used to classify test dataset into normal or malware labels. The actual and predicted results are compared using confusion matrix. The models are then evaluated for performance using cross validation parameters such as accuracy, precision, recall, f1-score and area under the Receiver operating characteristic curve (AUC-ROC). The comparative analysis of different models reveals significant differences in their performance metrics. The Random Forest Classifier outperforms other models with the highest accuracy of 0.99932 and an exceptional ROC-AUC score of 1.0, indicating its superior ability to distinguish between classes. The KNN model also shows strong performance, with an accuracy of 0.98816 and a ROC-AUC of 0.99740. In contrast, the Naive Bayes classifier demonstrates much lower performance across all metrics, with an accuracy of 0.39624 and a ROC-AUC of 0.7419, suggesting that it struggles with the dataset's complexity. The Decision Tree Classifier performs nearly as well as the Random Forest with an accuracy of 0.99804 and a ROC-AUC of 0.9988, though its precision score appears to be incorrectly reported, matching that of the Naive Bayes Classifier. Overall, the Random Forest Classifier and Decision Tree Classifier exhibit the best performance, making them the preferred choices for this classification task. In addition, the results of this research work are compared with results of other works in literature that used multiple datasets to train and evaluate a set of models. Their results were outperformed by this research work, even when they used more modern datasets.

CHAPTER ONE
INTRODUCTION

1.1 Background of the Study

In recent times, challenges confronting the world include infectious diseases (including COVID-19 Pandemic), environmental health hazards, hunger, and crime prevention (Sardi *et al,* 2020). One of the main challenges in the health sector is cyber threat. A threat could be defined as the combined probability of an unwanted event and its impact on the target object (Sardi *et al,* 2020). Another school of thought defined threat as potential event, intentional or otherwise, that compromises security attributes such as confidentiality, integrity, availability, authentication, access control and non-repudiation. (Tatam *et al,* 2021; HIPAA, 2024). Threat can also be defined as different kinds of likely actions that are caused by either stilt or natural means against a functional system (Idris *et al*, N.D).

A Framework is a graphical representation of the system being developed that shows the input, logical internal processes, and output (Tatam *et al,* 2021). While an electronic health system (EHS) is defined as a digital version of the paper-based system, which replaces the manual recording process with electronic healthcare records (EHR) (Hathaliya and Tanwar, 2020). EHS offers economic benefits such as efficiency, data management and administration of patients care system. It also enables the sharing of information between patients and their health care providers. This improves diagnosis, increases patient education and promotes personalized healthcare (Fatima and Colomo-Palacious, 2018). Also, EHS provides collaboration on patient care and emergency care. In addition, it gathers patients' data and stores in EHR.

However, EHS has become common targets for ransomware attack, crypto mining, data theft, phishing, spoofing and insider threats. Cyber-attacks are becoming sophisticated by the day with cyber threat actors exploiting the vulnerability that should have been patched before to avoid intrusions. A cyber threat actor is a person or entity that undermines security attributes. The neglect is not unconnected with non-adherence to security best practices, due to budgetary pressure, few skilled information technology (IT) professionals in the health sector, and ignorance or indecision of the most effective way to provide resilience to cyber threats (HIPAA, 2024). What is more worrisome is the scale of these data breaches, necessitating an urgent action to curb the menace.

Security is an important sector in information processing, especially in critical settings like EHS. Cloud Computing Technology (CCT) provides the enabling environment for the classification of threats at lower cost (Yeng *et al,* 2020). CCT is very useful to health care organizations, for it helps to focus on their therapeutic services. The benefits of using CCT for EHS include:

i. Easy collaboration and data sharing
ii. Mobility
iii. Cost reduction on information and communication technology (ICT) services
iv. "Pay for what you use", instead of owning structures and equipment
v. Scalability
vi. Business continuity
vii. Flexibility, and
viii. Strategic values.

These benefits not-with-standing, CCT does have shortcomings. The security challenges include (Fan *et al.,* 2024):

i. Data loss or leakage
ii. Transaction hijacking
iii. Insecure user interface
iv. Denial of service (DOS) attacks
v. Abuse of cloud services
vi. Malicious insider attack
vii. Data breaches
viii. Virtual machine (VM) escape
ix. Unauthorized access to management interface
x. Metering and billing systems
xi. Vendor lock-in

In vendor lock-in, a client can move its data and services to a cloud service prodder (CSP) but cannot easily change the vendor without severe cost, legal constraint or technical incompatibility (Yeng *et al.,* 2020). The menace of malicious programs threatening the healthcare system needs to be curbed, otherwise healthcare infrastructure may lose its relevance. For instance, healthcare and public health institutions had 23% ransomware attacks in 2021 (FBI internet crime report, 2021). In addition, healthcare is among the four organizations susceptible to multi-vector attacks (CBD report, 2021). Similarly, Verizon DBIR (2022) reported web application attacks and system intrusions as top cyber-attack patterns in healthcare industries. In 2023, the healthcare system suffered eleven (11) breaches in the United States of America (USA). These incidences took place in hospitals, insurance companies working with the

hospitals, and vendors working with medical facilities; up to 70 million people's medical records were affected (HIPAA, 2024).

The health sector depends on inter connected information systems, as such new security vulnerabilities are opened, making the industry a prime target for cybercrime. This puts patient data, underlying infrastructure and lives at risk. Traditional security measures often fail to keep pace with evolving threats. Some threats include phishing, spoofing, malicious insider, ransomware, data breaches, and more. These breaches, among others, are generally grouped into four attack types:

i. Denial of service (DOS) attacks
ii. Probe or surveillance attack
iii. User to root (U2R) attack
iv. Remote to local (R2L) attacks

These attack types can shatter the trust patients have for medical or healthcare institutions, if not controlled. Modern medical devices operate in networked clinical environment, and can attract data breaches, disruption to clinical operations, and invasion of patients' privacy and safety (Aijaz *et al.,* 2023). This research work intends to develop a framework that will train models, which will classify security threats in electronic health care systems into normal and abnormal types.

Machine learning (ML) tools such as Decision Tree (J48), Naïve Bayes (NB), Logistic Regression (LR), and K-Nearest Neighbor (KNN) will be trained by Random Forest, the algorithm, using National Science Laboratory-Knowledge Discovery in Databases (NSL-KDD) dataset. The trained models will learn the characteristics of normal applications and use that knowledge to classify malware from normal applications

(Abushark *et al.,* 2022). Abnormal applications also called malware include viruses, worms, Trojan horses, spyware, adware, ransomware, key loggers, among others.

Viruses are computer programs that attach themselves to executable programs (.EXE, .BAT, and .COM). They depend on the intervention of the user to spread. A worm, unlike the virus, is a computer program that does not depend on the user to spread. It replicates itself as it moves from one network to another. Trojan horse is a fake computer program that masquerades as a legitimate program but contains a malicious code that activates and runs when the need arises to attack the host system. Spy-ware or probe program appears to be dormant, and does not directly cause any harm on the host system. It rather stealthily gathers information, personal data, and vulnerable points of the affected host system or network and send back to the command and control (C&C) server. The data gathered usually include credit/debit card numbers, personal data like name, address, password, and so on. Adware are more or less inconvenient applications that pop-up on the screen unannounced, indeed, they are unwanted adverts. They are difficult to do away with.

The dataset to use in this research work is NSL-KDD dataset. It consists of normal and abnormal applications, grouped into DOS, PROBE, U2R and R2L. It is divided into training dataset (80%) and test dataset (20%) with forty-one attributes, including labels. Python programming language, in collaboration with its external modules like scikit learn, pandas, and so on, will be used to manipulate the dataset to produce results.

1.2 Statement of the Problem

In critical healthcare systems, network servers provide storage, communication channels and other services for the improvement of patient care using patient health records (PHR). Actors such as physicians, nurses, pharmacists, laboratory technicians

and patients access these assets in PHR using cloud computing technology However, the main challenges of PHR are security and patient privacy (Akram *et al.,* 2021). EHS security and privacy issues include (Singh and Chatterjee, 2019):

i. Unauthorized Access
ii. Impersonation attack through spoofing
iii. Malicious insider

1.3 Aim and Objectives

1.3.1 The Aim of the Study

The aim of this research work is to develop a framework using machine learning tools, which will classify malware threats from normal applications, in CCT platform.

1.3.2 Specific Objectives of the Study

The specific objectives of the study are to:

i. Build a framework using Random Forest algorithm to train base models such as Decision Tree (J48), Naïve Bayes (NB), Logistic regression (LR) and K-Nearest Neighbor (KNN) to learn the characteristics of normal applications and use that knowledge to predict labels of normal applications and malware attack types.
ii. Aggregate the models' predictions using VotingClassifier() and determine a consensus classifier called Hard vote classifier, which will be used to detect intrusions in test dataset.
iii. Use anomaly intrusion detection system (IDS) also called dynamic IDS, to analyze a specific point in the network or host system and detect threats; and set aside privileged users such as system administrators, computer operators or contractors.

These objectives are achieved in methodology.

1.4 Methodology

Electronic healthcare system and Patients Health Records (PHR) are central to patient care system. They are used to reduce morbidity and mortality of patients in hospital management. To curb the menace of security and privacy threats on them, the following steps are put in place:

i. Download and use NSL-KDD dataset to train and test the models for generalization. The dataset is gotten from Kaggle, a public data repository.

ii. Preprocess the dataset using selection techniques such as SMOTE, Pearson correlation (PC) and Information gain (IG); and then extract impurities using min-max normalization to reduce the features' dimensions to a range, say [0,1]; and remove redundant, noisy and missing data using principal component analysis (PCA)

iii. Convert symbolic (text) features such as protocol type, Services and flags to crisp values using One-hot-encoder() and label-encoder().

iv. Split the dataset into training dataset (80%) with 125,973 records; and test dataset (20%) with 22,544 records using train-test-split() function in a ratio of 80:20

v. Train the models such as DT, NB, LR in parallel using RandomForest(); and compute the shortest distance from k (number of nearest neighbors) in KNN to assign the target object.

vi. Predict the labels using each trained classifier with model.predict() function, then aggregate the predictions of the models and that of KNN into a consensus classifier, soft vote.

vii. Predict the labels using soft vote classifier

viii. Compare the predictions of the soft vote with the actual labels using confusion matrix (being a supervised learning problem)

ix. Generate results

1.5 Significance of the Study

The timeliness of this research work cannot be over emphasized, given the fact that electronic healthcare system (EHS) and its subsidiary, electronic health records (EHR) are very crucial in promoting health care, security and patients' privacy. EHR contains sensitive patient's data such as medical history, treatment records, and personal identification details such as name, address, password, credit card number, debit card number, and more. EHS and EHR deserve to be protected from malicious programs and hackers to avoid privacy breach, identity theft, and security compliance penalty.

Unfortunately, security and privacy attacks are constantly being launched against hospital management. Some of these challenges include

i. Unauthorized access, which could result from neglect and carelessness of physicians and computer operators

ii. Vulnerabilities of both hardware and software resulting from misconfiguration, system bugs and spyware attacks

iii. Data integrity, which could result from dynamic code injection, hijacking of processes, and phishing.

iv. Data collaboration between systems, organizations and governments, has the good side and bad side. Data collaboration has the advantage of sharing patients' records between authorized persons; however, the more data

moves between organizations so also are vulnerable points not only increasing but exploited by threat actors.

The solution to this problem is to adopt standardized protocols and ensure that data exchange formats are adhered to.

For instance, in 2018, OptimumHealth experienced a data breach caused by phishing, thereby exposing personal information of 10 million patients. Such information includes patient's name, address, date of birth, credit/debit card numbers and insurance information. (Singh and Chatterjee, 2019). Also, in 2015, Anthem, one of the health insurance companies in the US, experienced a hacking attack that affected over 78 million people. The hackers stole patients' information such as names, addresses, social insurance and medical information. These breaches highlight the need to protect EHR data from unauthorized access, disclosure, alteration or destruction. Some solutions to these breaches can be provided by the user, while this research work handles cyber security and patient's privacy to safe guard lives and property for the interest of stake holders, include:

i. Adoption of robust user authentication protocol such as two-factor authentication to reduce the risk of unauthorized access. Traditionally, password and user name are used, which are vulnerable and easy to break using brute force technique. Biometric identification is much harder to break than password.
ii. Update anti-virus programs to the latest version, this will include the latest vulnerability patches supplied by Google, for those using Android platform.
iii. Backup your data to reduce the risk of unforeseen data lose or host system crashing.

iv. Training both clinical and non-clinical staff of the healthcare organization to know what to do in the event of an attack; the significance of data privacy and the implication of data breaches.

v. Encryption protects the integrity and confidentiality of information stored in EHR.

In this research work, these challenges are addressed using machine learning ensemble classification techniques. The solution to these breaches will provide confidence and integrity in the patients, hospital management, insurance community and the general public. These are indeed, the stake holders in hospital management, with the objective of saving lives.

1.6 Scope and Limitations

This research work is delimited to the following intents:

i. Electronic Health System (EHS) has many security issues, but only unauthorised access, vulnerability, data integrity and confidentiality are considered; also considered is patient's privacy threat. Other security threats such as probe, non-repudiation, encryption and ransomware are not considered

ii. Security attributes covered in this research work include Confidentiality, Integrity, Availability, Other attributes such as confidence Index, Client App., Source IP, Date/Time, Object of Attack type, Object Name, and more, are not considered.

iii. The models are trained to learn the behaviour and characteristics of 67,343 normal applications, and use that knowledge to classify applications that

deviate from the norm as malware. Malware family signatures and their databases are not considered.

iv. Bagging technique, through Random Forest Algorithm, is used to train and combine multiple classifiers (models) such as DT, NB, SVM and KNN to solve the problem of malware threats. Other techniques such as hybridization, boosting and stacking are not considered.

1.6.1 Limitations

The framework in this research work cannot directly be deployed on desktop, PCs or mobile devices because of the following that will not be part of this research work.

i. The framework will not be integrated to application programming interface (API) for uploading to the cloud, and the front-end interface will not be designed and developed to enable linking with the backend.

ii. Other limitations include:

- Undocumented APIs and Applications from third party stores may still find their way to the device because Android devices accept them (70% of which are malware or faked applications).
- Dynamic analysis monitors one device component at a time in real-time for deviations from the norm by applications manipulating it and report as malware; other components are not considered at the time.

1.7 Definition of Terms

To flow with the current write-up, it is normal to define some of the terms that will be encountered in the course of reading the thesis.

Attack: In the context of computer and network security, an attack is an attempt to access resources on a computer or network without the necessary authorization or to bypass security measures put in place.

Artificial Intelligence: This is the ability of a computer to learn human expertise or cognitive function, builds that into a device to function or assist experts or users in this field. For example, some cars are capable of providing security for themselves devoid of human intervention.

Audit: This is to track security related events, such as logging onto the system or network, accessing objects or exercising user or group rights or privileges.

Backdoor: This is an opening or break in the software, hardware or network usually for debugging purposes. Malware code (Rootkit) is a malware that installs itself on the system and assumes the privileges of the system administrator, and paves the way for more malicious codes to be installed on the said system.

Botnet: This is a collection of internet connected devices and systems that have been compromised by malware infections. They constitute peer-to-peer (P2P) system with one of them serving as a proxy to send stolen information to the botmaster (or command and control server).

Breach: Successfully defeating security measures to gain access to data or resources illegally or illegitimately. Also, making data or resources available to unauthorized persons or delete or manipulate computer files.

Brute force attack: This is an attempt to crack passwords by sequentially trying all possible combinations of characters until the right combination grants access.

Buffer overflow: This is an attempt to crash the system by putting more data than the buffer can accommodate.

Buffer: A holding area for data

CIA triad: This stands for Confidentiality, Integration, and Availability of data and services.

Classifiers: These are trained models used to classify a binary data into normal or anomalous types

Confidentiality data: This ensures that the content of data is kept secret

Counter measures: Steps taken to respond to an attack or anomaly.

Cracker: A hacker who specializes in cracking or revealing or discovering system passwords in order to gain access illegally to a computer or network system.

Crash: Sudden failure of a computer system, rendering it unstable.

Data availability: This has to do with reliable and timely access to data (24/7), anytime, anywhere.

Defense in dept.: The practice of implementing multiple layered security.

Denial of service attack: A deliberate way of bashing the target system or resource with request, so much such that legitimate users would be starved of its services.

Exposure: A measure of the extent to which a computer or network is open to attack, based on its particular vulnerability, how well known it is to the hackers, and the time of operation.

Hacker: An expert computer programmer, who spends his or her time writing programs that do negative intents of his or hers.

Integrity of data: This is a security characteristic which ensures that the data sent or received is not modified.

Intelligent: The intelligence in this case is referring to systems or devices learning ability of human expertise, and using that knowledge to assist experts in that domain.

Intrusion Detection: This is the process of monitoring host systems, mobile devices or network for abnormal behavior of applications manipulating the device component. Intrusion is an unlawful entry into the device or unlawful manipulation of a device.

Least privilege: The job schedule of an employee, the level of access to the system resources and information granted to such an employee.

Machine learning: This is the process whereby computers learn features of data or human expertise and uses that knowledge to recognize similar features that may not be labeled.

Malicious code: A computer program or script that performs actions inimical to the system, but satisfies the master's evil intent.

Mobile Device: This is a handheld device used to communicate, and do other things like buying and selling online, pay bills, make transfers, using the internet.

Penetration testing: Evaluating a system by attempting to circumvent the computer's or network's security measures.

Reliability: The probability of a computer system or network continuing to perform satisfactorily under normal operating circumstances.

Risk management: The process of identifying, controlling, and mitigating the problem militating against data confidentiality, data integrity, and system availability.

Risk: The probability that a certain security threat will exploit the system's vulnerability (weakness) resulting in damage, loss of data or other undesired effects.

Sniffer: Packet sniffer is a program that captures data in transition across the network, for nefarious intents, in most cases.

Social engineering: This is a process whereby malicious programs persuade users to either click an infectious site, download or install a program, the consequences in most cases, he or she may not know.

Threat: This is a potential danger to data or system or even the network. Threat agents are employed to cause the threat, they include: Trojan, virus, worms, spyware, adware, etc.

Vulnerability: This is a flaw or bug or weakness in the software or hardware component of a system that can be exploited by the "bad boys" to their advantage.

Zero-day vulnerability: This is an undisclosed defect or flaw, which attackers can exploit. It is called zero-day because it is not publicly reported before becoming active.

Zombie: This is a compromised computer, which is usually connected to the network, the malware agents that infect such system could be a virus, worm, Trojan, etc. It can be used to perform malicious tasks, and in most cases made to join the botnet family.

1.8 Organization of the Thesis

In this research work, chapter one introduces electronic health system (EHS) and one of its subsections, electronic health records (EHR) among other sub types of EHS. Their cyber problems, and in particular security and patients' records privacy motivated the researcher to design and implement a framework that will classify the threats ravaging the health sector. Chapter two presents the detailed literature review, especially, specific words (terms or phrases) in the thesis title and other related topics in literature. The chapter also reviews related literature and compares their strengths and weaknesses in relation to the current work being developed. In chapter three, the method used to develop the framework, which is ensemble learning technique, is discussed. Five classifiers are trained and aggregated to produce a consensus model, called soft vote, which is used to classify the test dataset for generalization and performance. Random forest (RF), decision tree (DT), K-nearest neighbor (KNN), Naïve Bayes (NB) and logistic regression (LR) are the classifiers used in the development of the framework. Chapter four presents the results of the threat detection and the performances of the

trained models, and that of the soft vote classifier. Confusion matrix is used to compare the expected labels and the predicted labels to ascertain the true positive rate, the true negative rate, the false positive and false negative values of each model. A discussion was also put in place that compares the relationship of the models in terms of accuracy, precision, recall, f1 score and area under the curve (AUC) of ROC values. In chapter five, the work is summarized, concluded and our contribution to knowledge duly stated. The problems encountered during the framework development are reduced to future h works.

CHAPTER TWO
LITERATURE REVIEW

2.1 Preamble

Electronic Health System (EHS) is the digital use of information and communications (ICT) to support health and health related fields. It includes health information techniques such as (Alobo *et al.,* 2020):

i. Patient health information and data
ii. Clinical decision support system
iii. Results management and central data repository
iv. Computerized physician order entry (CPOE)

These health systems are used to support the organization in the care and delivery of health services (Alobo *et al.,* 2020). For instance, they improve the use of physician time, improve patients' safety, improve efficiency in health management, enable the exchange of information, patient data and medical images electronically, and assist physicians in reducing patients' morbidity and mortality rates.

EHS is applied in the following areas:

i. National Health Insurance Scheme (NHIS) management
ii. Cashless payment integration
iii. Electronic identification of patients
iv. Bed assignment
v. Laboratory orders, reporting and consumables
vi. Easy and quick retrieval of patient's records

The challenges of EHS include:

i. Threat to patient's privacy
ii. Information overload
iii. Power outages, among others.

In addition, electronic health records (EHR) is the digital collection of patients records in a central database, which can be shared across different health settings to provide collaboration, real-time decision support, and permanent patient health records. Although these capabilities improve health care quality, their vulnerability to malicious attacks (threats) has become worrisome. If something is not done to curb these threats, health care infrastructure will lose its relevance, patients' trust and patronage (Aijaz *et al.,* 2023).

The importance of security and privacy of patients in health care and the need to preserve life, reduce morbidity and mortality rates has motivated this researcher to design and implement a framework that will classify threats in critical lifesaving electronic healthcare system.

2.2 Theoretical Framework

Traditionally, in Africa, and Nigeria in particular, healthcare is paper-based, patients records are paper based, referral of patients from one hospital to another is paper based, even communication between units within a hospital is paper based. In the 21st century, where there are many means of automated systems like smartphones with smarter means of communication such as SMS, MMS, electronic mailing, and social media, there should be a change in medical health system. The current paper-based system is fraught with problems such as:

i. Scalability is not tenable
ii. The paper-based system is prone to error and damage by pests
iii. The files are degraded with time
iv. The system is highly unreliable in a critical setup like the medical healthcare system
v. Retrieval is time consuming
vi. There is no visible audit trial
vii. The file cabinets and shelves occupy too much space.

The researcher is motivated by these shortcomings to develop and implement a framework that will monitor and report critical security threats. The framework will be automated devoid of human intervention.

The dataset to be used in this implementation is national science laboratory – knowledge discovery in databases (NSL-KDD) dataset. This dataset will be split into two parts, training dataset (80%) and testing dataset (20%) and used to train and test the classifiers respectively. Machine learning technique such as ensemble learning (bagging) will be used to implement the framework, with cloud computing technology (CCT) as the platform.

Algorithms to train using the dataset include Logistic regression (LR), decision tree (DT), K-nearest neighbor (KNN), random forest (RF), and naïve Bayes (NB).

The trained models will be used to predict labels. Their predictions will be aggregated using soft vote classifier to produce a consensus model that will be used to predict labels using test dataset, which they did not train with. The predicted labels will be compared with expected labels using confusion matrix, being a supervised learning problem.

The trained models' performances will be evaluated using the following parameters, accuracy, precision, recall, f1-score and area under the curve (AUC). Python programming language and its external modules such as scikit learn, pandas and NumPy will be used to train, test and evaluate the models.

The significance of the research work is to prepare the minds of hospital management on electronic healthcare system, which is more useable, accessible, accommodating and versatile, given the modern-day communication networks, including social media.

2.3 Electronic Health System (EHS)

EHS is the use of Information and Communications Technology (ICT) to support health and health related fields. It encompasses a wide range of uses such as telemedicine, smartphone applications, sensors and wearable; from mobile health (m-Health) to telehealth, and increasingly underpins all health-related activities (Singhal and Cowie, 2020). Other e-health subsidiaries include Electronic Health Records (EHRs), Sensors, non-invasive sensors, wearable, invasive sensors, tele-health, self-care and personalized care.

EHR offers varying levels of functionality such as basic documentation, data entry, real-time display of clinical signs and observations, communication and interoperability with other healthcare professionals. Indeed, there is no unique EHR within any country; individual healthcare organizations purchase their software and use, even find it difficult to interchange records for fear of loss, theft and mutilation of records (Singhal and Cowie, 2020).

m-Health is the use of mobile wireless technology like smart phones for remote access to healthcare information services. Sensors are integrated to m-health systems. They measure signals and collect data that are transmitted or recorded for analysis. Sensors

are subdivided into evasive and non-evasive sensors. Wearable and non-wearable are examples of non-evasive sensors. Telehealth is the use of telecommunications and virtual technology to deliver health care outside of the traditional healthcare settings.

Nykanen (2017) defined e-health as a socio-technical system composed of healthcare organizations, service providers, professionals, customers, citizens and patients; two-sided market, industrial companies providing their products and services, technology-mediated platforms for communication, and infrastructures that in collaboration provide value and services. Infrastructure and services are needed for knowledge sharing, management, and exchange of information and data.

An e-health ecosystem in two-sided market means that products and services are developed and delivered to meet the customers' needs. Customers are health professionals, patients, citizens, service providers, healthcare organizations and suppliers.

However, Eysenbach (2001) defined e-health as an emerging field in the intersection of medical informatics, public health and business, referring to health services and information delivered or enhanced through the internet and related technologies. In a broader sense, the term e-health involves many things; technical development, state-of-mind, a way of thinking, an altitude, and a commitment for networked global thinking to improve healthcare locally, regionally and globally using ICT. Indeed, e-health encompasses more than internet and medicine, as such the "e" in e-health does not only stand for 'electronic" but implies a number of other "e's" such as depicted in Table 2.1 (Eysenbach, 2001:

Table 2.1: The Meaning of "e's" in e-Health System

S/N	"e's"	What the "e" stands for
1	**Efficiency**	e-health increases efficiency in healthcare by reducing cost, avoid duplication of diagnostics or therapeutic interventions between healthcare establishments, and patient's participation.
2	**Enhancing** quality of care	Increasing efficiency does not only reduce cost but enhances quality of service and healthcare.
3	**Evidence** based	e-health should provide scientific analysis in the laboratories to test for suspected ailments especially in diseases with confusable symptoms.
4	**Empowerment** of consumers and Patients	e-health makes the knowledge bases of medicine and personal electronic records accessible to consumers and patients over the internet hence providing patient centered medicine.
5	**Encouragement**	Patients and healthcare professionals' relationship is encouraged towards true partnership in decision making.
6	**Education**	Physicians, patients and other health workers should be educated in the use of ICT gadgets and medical terms used, diagnostics, which are tailored towards preventive and personalize healthcare.
7	**Enabling**	Information and record exchange between health care workers, government agencies and insurance companies in healthcare be enabled through safe ICT transmitting media.
8	**Extending**	The scope of health care is extended beyond the conventional boundaries, through global health providers online.
9	**Ethics**	The patient-physician interaction poses new challenges and threats to ethical issues such as online professional practice, informed consent, privacy and equity.
10	**Equity**	e-healthcare should be equitable between the "haves and have-nots". Political measures and legislation should ensure that the digital divide running between the rural vs. the urban populace, rich vs. the poor, young vs. the old is bridged.

Source: Eysenbach (2001)

In addition to the 10 “e’s”, e-health should also be easy to use, entertaining exciting and, indeed exist. In EHS, the patient medical data are stored remotely with cloud service providers (CSP) in servers and huge data bases as depicted in Figure 2.1

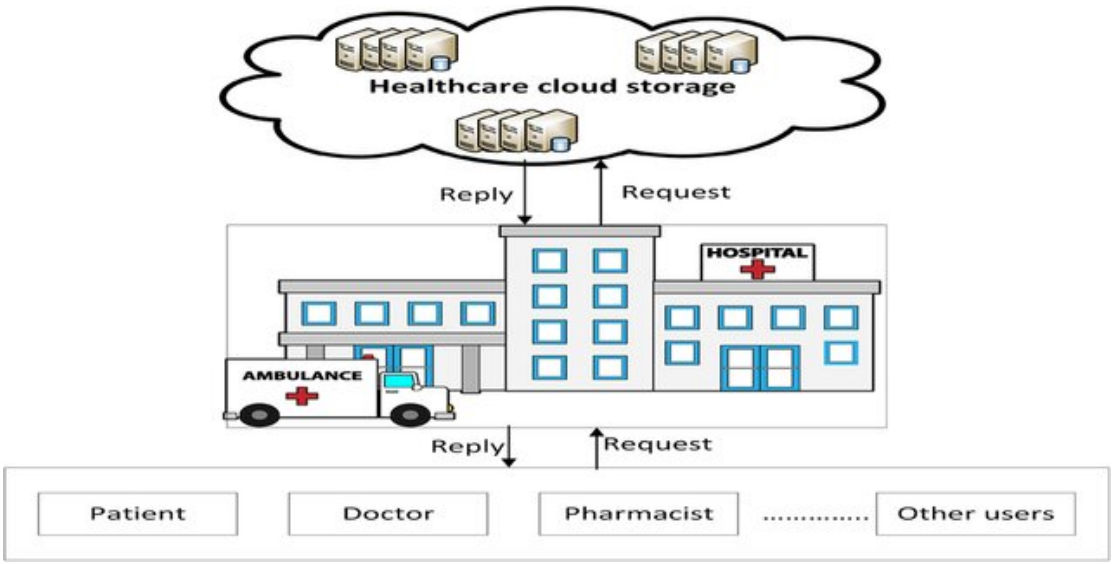


Figure 2.1: Electronic Health System with Patients’ Records Stored in the Cloud
Source: Singh and Chatterjee (2019)

Some CSPs include:

i. Amazon AWL

ii. Microsoft Azure

iii. Google Cloud

iv. IBM

v. NetSuite.

From Figure 1, the CPS collaborates with the hospital, which is responsible for managing the access rules, authentication mechanism and providing security and privacy to the patients’ data (Singh and Chatterjee, 2019). To treat a patient, the physician and other healthcare workers request for the patient’s record through the hospital using the healthcare application. The patient can also access his or her personal medical records. Delegate users can also obtain permission to access patient’s medical

records or issue permission to the healthcare practitioners to access their data. Other healthcare users such as researchers, insurers or government agencies can also access the patients' medical records through the hospital. In case of emergency, some access rules are relaxed to accelerate privilege treatment. However, EHS and its subsidiary HER are not fully adopted by patients, citizens and medical practitioners.

Health workers and patients have varying reasons for not wholly accepting EHS/HER. Some of the reasons include:

i. Lack of awareness of, and confidence in healthcare solutions among patients, citizens and healthcare professionals.
ii. Lack of interoperability between e-health solutions.
iii. Limited evidence on the cost effectiveness of e-health tools and services
iv. Lack of legal clarity for health, and wellbeing of mobile applications
v. Lack of transparency regarding the utilization of data collected by such applications.
vi. Fragmented legal frameworks, including the lack of reimbursement schemes for e-health systems.
vii. High start-up cost in setting up e-health systems or business
viii. Regional disparity in accessing ICT services, especially deprived areas

On the whole, education will go a long way to disabuse the minds of patients and e-healthcare practitioners to enable patients, young or old to access e-health interventions.

2.4 Electronic Medical Records (EMR) in Healthcare Systems

An Electronic Medical Record (EMR), also called Electronic Health Records (EHR), is one of the subsidiaries of EHS. EMR is an electronically stored digital version of medical data, where patients' examination results, prescribed medication, as well as

treatment history records are stored (Wesolowski *et al.,* 2016). EHR are real-time records, which are patient-centred and accessible to all authorized personnel including health care providers and organizations (Wesolowski *et al.,* 2016). By patient-centred healthcare we mean that the patient is given his or her dignity when treated, respected and involved in decision taken on his or her healthcare needs It empowered the patient to become active participant in his or her care.

For example, physicians, nurses, laboratory technicians, pharmacist and patients are actors manipulating the asset (EHR). Although EHR and EMR are seemingly used interchangeably, there are slide differences. EMR allows for electronic data entry, storage and maintenance of digital medical data, while EHR contains the patient's records from doctors (physicians), which include demographics, test results, medical history, History of Present Illness (HPI) and medications. Therefore, EMRs are part of EHRs and may contain the following (HHS, 2022).

i. Patient registration, billing, preventive screening or check-ups.
ii. Patient's appointments and schedules
iii. Tracking patient's data over time
iv. Monitoring and improving overall quality of care

However, EHRs are threatened by malware and the threats could be categorized into

i. Human factors: employees, contractors or hackers.
ii. Natural disasters, such as fire, earthquakes and flood
iii. Technical failures such as system crash, vulnerabilities, malware attacks, and hardware misconfigurations.

Human threats could be further categorized into internal or external; intentional or accidental. For instance, in 2015 up to 113 million breaches were recorded

(Wesolowski *et al.,* 2016). The ubiquity of computers and mobile devices has increased the risk of unauthorized access to EHR data. These risks are based on a range of factors, which include:

i. User related issues

ii. Financial issues

iii. Design flaws

These risk factors prevent EMR/EHR from being used as effective tools to deliver healthcare services. In recent times, it has become a common target to health care breaches. In 2020, at least 2,354 US government healthcare facilities and schools were breached by ransomware (HHS, 2022). This attack caused significant disruption across the healthcare industry. Also, in 2021, Health and Human Services (HHS) received reports of data breaches from 578 healthcare organizations, impacting more than 41.45 million individuals. Figure 2.2 depicts top threats against EMR/EHR systems.

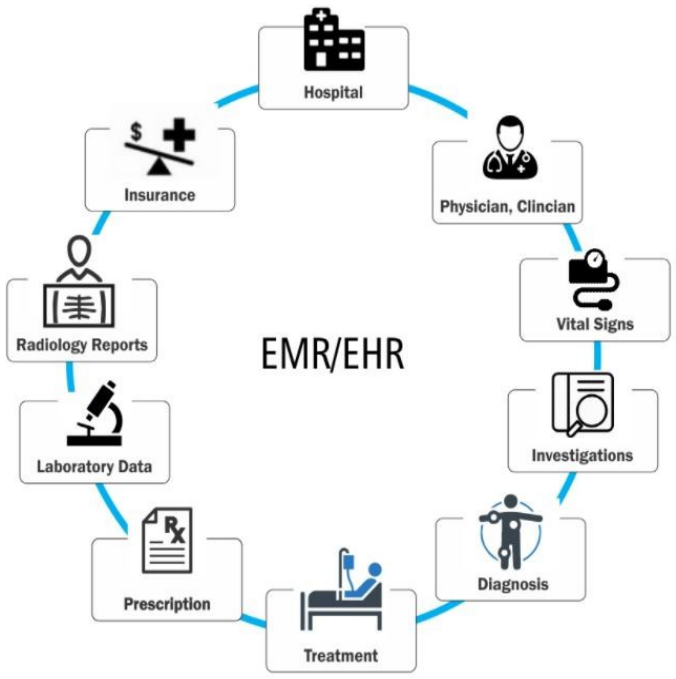


Figure 2.2: Top Threats Against Electronic Medical and Health Records
Source: HHS 2022

The threats impacting Figure 2 include:

i. Phishing attack

ii. Malware and Ransomware attacks

iii. Encription Blind spots

iv. Cloud threats

v. Employees (staff, contractors, supliers)

A phishing attack is a social engineering scam, where threats actors pretend to be trusted sources and lure innocuous user to open the email or click a link. A social engineering attack depends on the user to propergate. Figure 2.3 depicts a phishing attack.

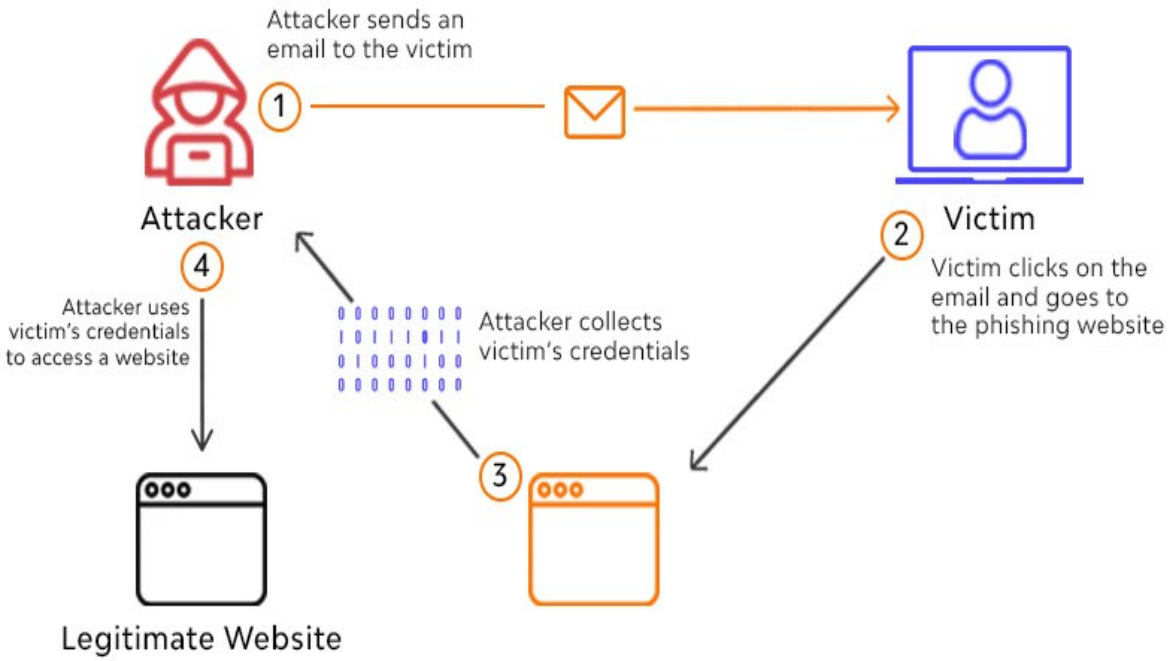


Figure 2.3: Phishing Attack on Electronic Health Records
Source: HHS (2022)

Ransomware is a type of Trojan malware (malicious software) that locks out the user from their host system or network and demands for a ransom, usually paid in bit coin. For hospitals, this could result in death as it disrupts patients' care. Figure 2.4 depicts stages of ransomware attack

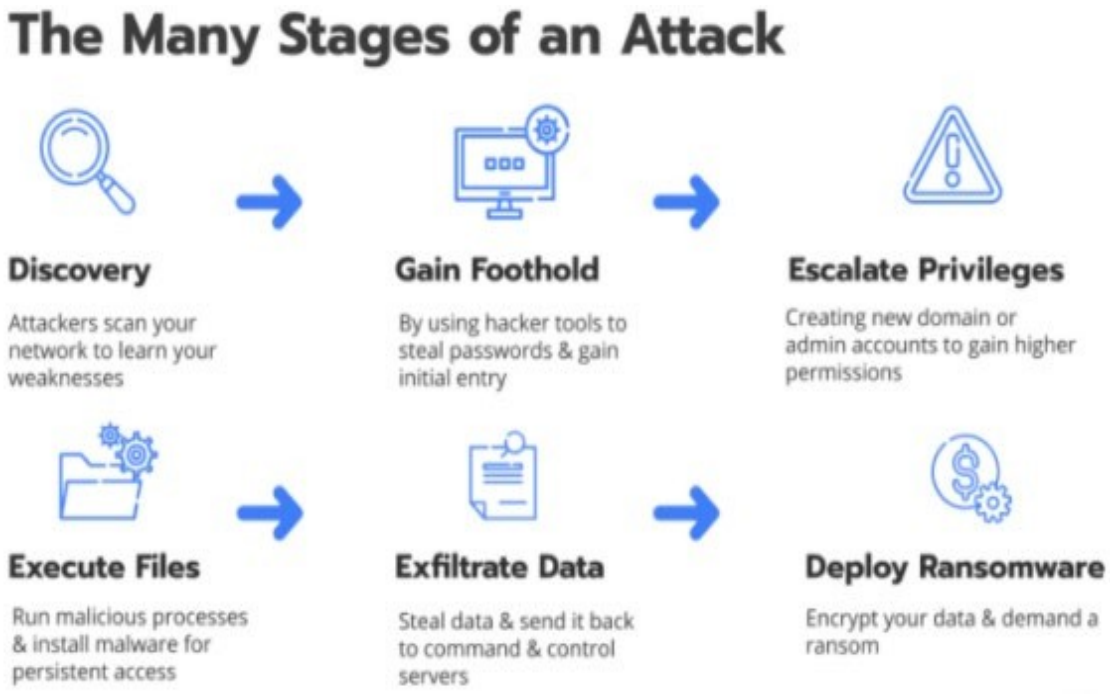


Figure 2.4: Stages of Ransomware Attack on EHR
Source: HHS (2022)

Encrypted blind spots are exploited by hackers to hide and avoid detection until they execute their payload on the target object.

The breaches in healthcare systems include:

i. Security and privacy issues
ii. Vulnerability
iii. Data can be lost or stolen or destroyed
iv. Inaccurate paper to computer transcription
v. Cause of treatment error

These breaches are propagated by hackers because EHR/EMR data have economic value in the black market (HHS, 2022). The data components or features include:

i. Name
ii. Geographic data
iii. Email address
iv. Health plan beneficiary numbers
v. Web URL
vi. Full face photo and comparable images
vii. Date
viii. Telephone numbers
ix. Fax number
x. Medical Record number
xi. Device identity and serial number
xii. Biometric identity such as retina scan or fingerprint
xiii. Social security number
xiv. Account number
xv. Internet Protocol (IP) address

From the foregoing, it is important to educate healthcare professionals to verify all EHR requests before sharing files with any remote request. The fight against malware threats is carried out in the cloud, which provide several services in many platforms. Given its advantages, many industry owners are now hoisting their data and services in the cloud. Figure 2.5 depicts the services of cloud computing technology.

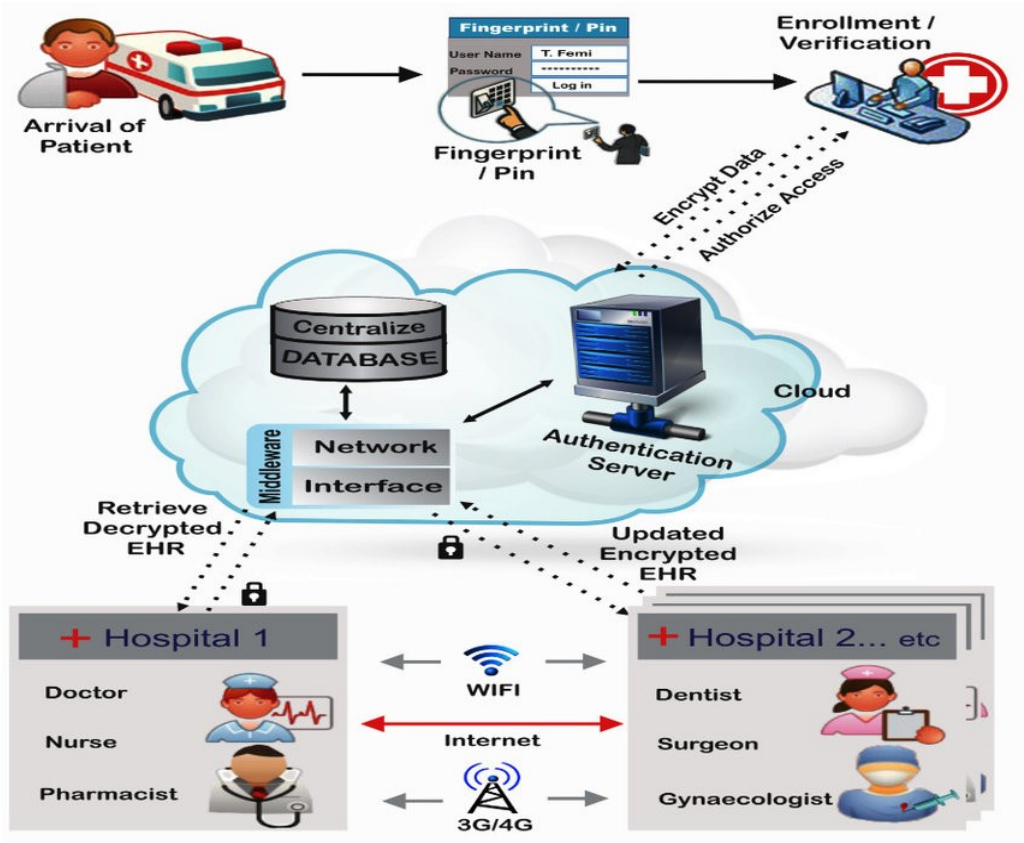


Figure 2.5: Cloud Computing Services in Medical Health Care System
Source: HHS (2022)

From Figure 2.5, the patient arrives in the hospital and undergoes the process of registration, identification, verification if he or she is regular patient, and activities of the various healthcare personnel as they manipulate the patient data in the cloud.

However, to reduce internal threats in healthcare organizations, the cyber security strategy and policies must be adhered to, such as:

i. Educate all healthcare patients and staff
ii. Enhance administrative control
iii. Monitor physical equipment and systems across the organization,
iv. Create workstations policies such as auditing and monitoring system users, employ device media control, and apply data encryption.

2.5 Security

Security is a process that protects systems, data, and transmission routes from unauthorized access (Pawar and Anuradha, 2015; Yesilyurt and Yalman, 2016; GSMA, 2019). Security has the following attributes: confidentiality, integrity, authenticity, non-repudiation, availability and access control.

i. Confidentiality prevents unauthorized access to data. (Babita and Kaur, 2017).

ii. Integrity prevents unauthorized modification or manipulation of data in transition.

iii. Authenticity verifies the identities of communicating parties in a transaction.

iv. Non-repudiation ensures that communicating parties do not disown their involvement or commitment to the transaction.

v. Availability ensures that required resources and services are available to authorised parties. It prevents denial of service or denial of access to legitimate users.

vi. Access control regulates who uses resources in a computing environment. It ensures that only legitimate communicating parties have access to data, through user name and password, digital signature, and finger printing.

There are many types of computer security such as physical security, network security, and cyber security. In this research work, physical security and network security are discussed. Figure 2.6 illustrates the security attributes as modified from Babita and Kaur (2017):

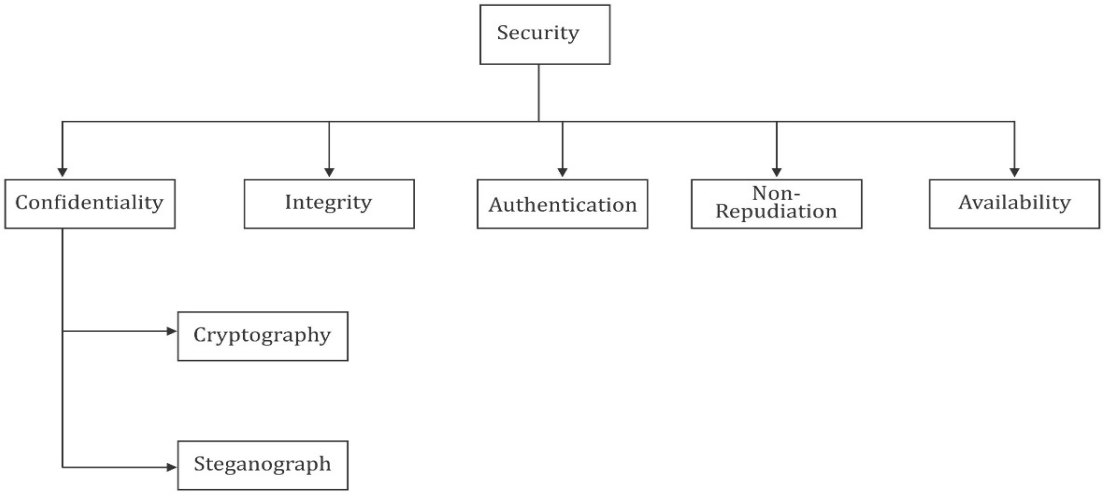


Figure 2.6: Security Attributes

2.5.1 Physical Security

Pphysical security of network environment is designed to protect networked computers, servers, routers, switches, hubs and gateways that house important data and system files. Unfortunately, this aspect of network security is often ignored by system administrators, and more often than not exposed to attack (Pawar and **Anuradha** 2015; Stein, 2020). Figure 2.7 depicts a vulnerable hub in a network system. For instance, Ethernet hub broadcasts data through every port; if a spare port is left unused, it becomes vulnerable as an intruder can plug his or her laptop in such a port and as the hub sends out data to legitimate systems, the same is sent to the intruder. Packet sniffing can also be plugged in such a port for the same purpose; an example is address resolution protocol (ARP) spoofing (faking the ownership of a particular address). Internet control message protocol (ICMP) uses routers to broadcast messages that contain their IP (internet protocol) addresses, which could be intercepted if left unattended to. These examples serve to demonstrate the fact that physical security measures should not be ignored by company management.

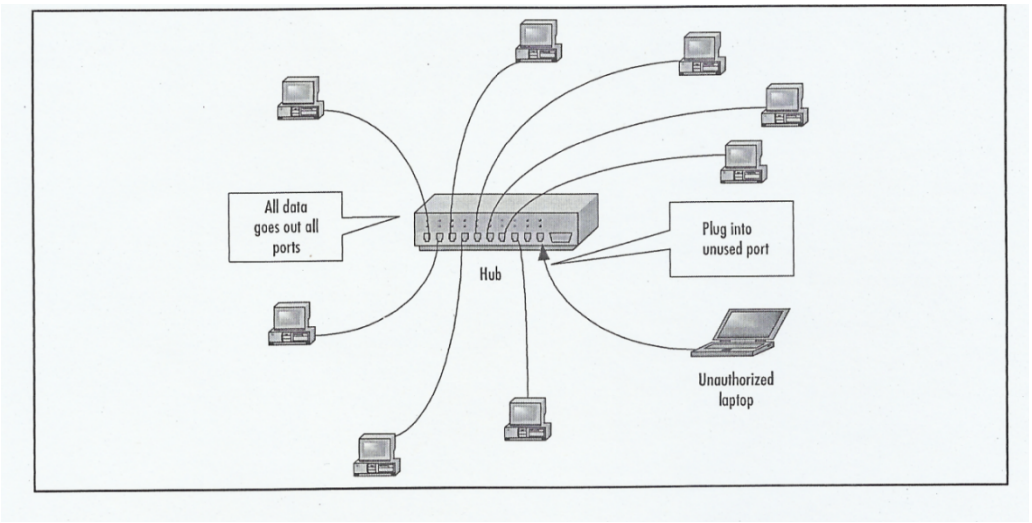


Figure 2.7: A Vulnerable Hub in a Network System
Source: Stein (2020)

Ethernet unshielded twisted pair (UTP) and coaxial cables are also vulnerable to data capture by tapping into them using appropriate devices (tools) if not attended to. Fibre optic cables that use laser light pulses to transmit binary data (1s and 0s) are not free from attack either. Optical splitters are used to tap into optical fibre cables for unauthorized access to data (Pawar and Anuradha 2015; Stein, 2020). Although physical security (wired medium) has its shortcomings, it complements other network security types.

2.5.2 Network Security

Network security consists of the six attributes: confidentiality, integrity, availability, authenticity, non-repudiation, and access control using firewall, encryption, steganography, intrusion detection system (IDS), intrusion response system (IRS) and more (Pawar and Anuradha, 2015). These security techniques are applied to systems or data features using network transmission media, which are either wired or wireless.

Three varieties of wireless media abound, such as radio (narrow band or spread spectrum), satellite (microwave), and laser (infrared). For instance, radio-based media operate according to IEEE 802.11i standards. However, wireless media have many

drawbacks in network security. They are susceptible to eavesdropping and man-in-the-middle (MITM) attacks that hijack (swindle) legitimate transactions from their owners. Laser signal may not be that susceptible, but it is a line-of-sight technology and sensitive to ambient factors like weather, flood, and earthquake (Pawar and Anuradha 2015; Stein, 2020).

For instance, in 2018, there were 53,308 security incidents, 2,216 data breaches, and 65 countries involved in these data breaches (Verizon, 2018). From these security breaches: 76% attacks were financially motivated, 73% of cyber-attacks were perpetrated by members of organized crime (outsiders), while 28% of such attacks were perpetrated by insiders (Verizon, 2018; Cheng *et al.,* 2017; Sharma *et al.,* 2020). In the public sector, 304 confirmed breaches were reported in 2018, they include cyber espionage, privilege misuse, web applications and lost or stolen assets (Verizon, 2018; Vemprala and Dietrich, 2019; GSMA, 2019).

Every organization, and individuals alike, needs a secured network as unstructured data is generated from there, stored there, and processed there. Unstructured data do not have a pre-defined data model nor organized in a pre-defined manner; examples include social media comments, images, emails, and rich media. Unstructured data is generated and transmitted using communication networks which include wired and wireless technologies. Wired technology is the communication media that involves Ethernet unshielded twisted pair (UTP) cables, coaxial cables and fibre optic cables. These cables are used to transmit data from source to destination (Ibrahim, Danladi, and Aderinola, 2017).

Wireless technology uses electromagnetic or acoustic waves such as radio (narrow band or spread spectrum), satellite (using micro waves), and laser (propagating infrared) to transmit data over short and long distances between sender and receiver (Ibrahim *et al.*,

2017; GSMA, 2019). The advantages of this type of transmission include reduction to cable distance, dynamic network formation, low cost and ease of deployment. The areas of application include Bluetooth, Wi-Fi (an IEEE 802.11 protocol), global positioning system (GPS), general packet radio service (GPRS), two-way radio, television remote control, mobile devices, wireless modem, computers, wireless keyboard and wireless mice (Yoti and Saini, 2017; Ibrahim *et al.*, 2017). Table 2.2 depicts the comparison between wired and wireless networks.

Table 2.2: Comparison of Wired and Wireless Networks
Source: Yoti and Saini (2017)

S/N	Characteristics	Wired networks	Wireless networks
1	Installation	Difficult to moderate: more components to connect using cables.	Easy installation
2	Visibility: Node to node	All nodes on a wired network can hear each other	Many nodes on a wireless network cannot hear each other on the same network
3	Visibility: Network to Network	Wired networks are invisible to other wired networks.	Wireless networks are often visible to other wireless networks. They do affect the performance of each other.
4	Cost	Low cost	High cost
5	User connectivity	Limited	Connectivity is wide
6	Mobility	Limited	Outstanding
7	Reliability	High	Reasonably high
8	Efficiency	High	Very high
9	Speed and bandwidth	High, up to 100mbps	Low, up to 54 mbps
10	Type	LAN, WAN, MAN	WLAN, WMAN, WWAN, WPAN; GSM, TDMA, CDMA; Wi-Fi (802.11) network, Bluetooth, Infrared network
11	Signal loss and fading	Less interference	More interference due to absorption, refraction and reflection
12	Interference	Less	High (radio, weather, other wireless devices and obstructions like walls
13	Quality of service	Better	Poor- delay and longer connection time.

2.5.3 Wireless Network Protocols and Their Vulnerabilities

There are many wireless protocols controlling networks. Protocols are sets of rules that govern the relationship between computers and other equipment within the network. They, however, are vulnerable and permeable to attacks. Some of them include:

i. **Access Point (AP**) protocol

This is a hardware device that enables the connection of wireless communication devices such as smart phones, tablets, PDAs, and laptops to a wireless network. Usually, an AP connects to a wired network and provides a bridge for data communication between wired and wireless devices.

ii. **Open System Authentication (OSA) protocol**

OSA is an IEEE 802.11 standard. It enables a handshake between sender and receiver. The client sends authentication request (in plain text) to the server, which responds in cipher text. However, authentication management frames are sent unprotected (in plain text). Upon a successful handshake, both stations (source and sink) are considered mutually authenticated. For a better communication security, OSA can be hybridized with wired equivalent privacy (WEP) protocol. WEP is used to encrypt the plain text, only after the "hand shake". This communication between sender and receiver can be tapped by hackers since it is done in plain text. However, steganography is the solution (or a hybridized form with cryptography).

iii. **Shared Key Authentication (SKA) protocol**

SKA uses WEP (wired equivalent privacy) and a shared secret key to provide authentication. Upon encryption, the authentication client will return the cipher text to the AP for verification. Authentication is a success if the AP decrypts the cipher text back to same plain text.

iv. **Service Set Identifier (SSID) protocol**

SSID allows wireless clients to communicate with appropriate AP. With proper configuration, only clients with correct SSID can communicate with the AP. SSID acts as a single shared password between the AP and the clients.

v. **Wired Equivalent Privacy (WEP) protocol**

WEP, an IEEE 802.11 standard, is used to provide confidentiality to data transmission over a wireless network, by encrypting the said transmitted data or information. Hackers have cracked WEP, and made it insecure, using automated tools, which are freely available on the net for download.

vi. **Wi-Fi Protected Access (WPA 1 and 2)**

WPA 1 is a wireless security protocol designed to fix known security flaws in WEP and improve authentication. However, WEP 2, which is based on IEEE 802.11i, is a revised protocol which allows only authorized users to access a wireless device (Ibrahim *et al.,* 2017; Yoti and Saini, 2017).

The vulnerabilities of these protocols are exploited by malware intrusions.

2.4.4 Cyber Security in Electronic Healthcare Systems

Cyber security refers to the measures that are taken to protect data and information stored on all electronic devices that are connected to the internet from malicious software (malware), hacking operations, data theft, sabotage, and transmission disruption (Saeed and Asaad, 2022). To curb the menace of malware threats involves traditionally static analysis and dynamic or anomaly detection methods.

Static analysis methods decompose or decompile the application to learn its characteristics and determine its payload physically, without running the said application. While these techniques are efficient and detect known malware using their family signatures stored in databases, they are however, easily avoided using

obfuscation techniques, polymorphism and metamorphism. They cannot detect new variants of the known malware and zero-day attacks. Zero-day attacks affect vulnerabilities that have not been made public before being exploited by hackers. Static analysis main drawback is that they do not access the source code of the programs they analyze, more so, it is difficult to extract binary code (Kumar and Das, 2023). Unfortunately, majority of security solutions depend on signature families, and are therefore static. To address this pitfall in static analysis, dynamic solutions are embraced (Saeed and Asaad, 2022).

Anomaly detection methods are aimed at identifying deviations from normal system's characteristics or user behavior. However, they are prone to false positive rates (FPR), can be difficult to tune and maintain (Katiyar *et al.,* 2024). Again, anomaly-based detection of malware threats operates in silos (confined environments like sandboxes), focusing on specific aspects of the network or host system, such as endpoints, servers, or applications without a holistic view of the whole security setup. This fragmentation approach leads to blind spots and oversight in detecting threats that span multiple domains (Katiyar *et al.,* 2024).

The limitations of these traditional techniques, coupled with increasing volume, velocity, and variety of cyber threats, necessitated the use of machine learning techniques, on aspect of computational intelligence (soft computing) to solve such problems. Machine learning (ML) approaches have the following abilities:

i. Improved threat detection by analyzing huge data sets (big data) to identify patterns and anomalies and unravel the Mistry of unknown and zero-day threats.
ii. It carries out automated discovery of threats without human intervention (system administrators).

iii. ML algorithms can continuously adapt and learn new threats and are scalable when new datasets are added to train, compared to rule-based approach

iv. ML models learn the characteristics of normal applications and use that knowledge to predict the type of an application presented to it, to enable mitigation.

To effectively use ML tools, the data input features need to be cleaned, normalized and reduced to acceptable principal components using principal component analysis (PCA).

The process of transforming the raw data into a usable format is called feature engineering. Feature engineering involves the extraction of statistical properties, header information or byte sequences form the network traffic data. These parameters are analyzed and the results gotten are used to compute the performance of the models. Common performance evaluation parameters are accuracy, precision, recall, f1-score and area under the curve (AUC), an aspect of receiver operating characteristic (ROC) curve.

To achieve these parameters, ML tools such as Logistic regression (LR), decision tree (DT) (decision tree can be C4.5 or J48 or ID 3), Gaussian naïve Bayes (GNB), K-nearest neighbor (KNN), random forest (RF) and deep-learning neural network (DNN), among others. They are usually trained to classify malware threats and then tested for generalization with new or dataset that it did not train with. In supervised learning problem, the predicted results are compared with the expected (known) values or labels using confusion matrix.

However, these activities are carried out in the cloud using cloud computing technology, with facilities provided by cloud service providers (CSP).

2.5.5 Cloud Computing Technology

The initiative of cloud computing technology (CCT) is based on the principle of reusability of ICT capabilities (Okediran *et al.,* 2022). More so, the emergence of CCT led to the reduction in cost of infrastructure, computation, application hosting, content storage and delivery using the "pay for what you use" policy, rather than owning infrastructure at such high cost. Cloud computing technology (CCT) is a distributed system (DS) aimed at providing unlimited shared resources to registered users. CCT provides physical implementations such as (Azeez and Der Vyver, 2018):

i. Software as a service (SaaS)
ii. Platform as a service (PaaS)
iii. Infrastructure as a service (IaaS)

Also, the implementation aspect of CCT include:

i. Private cloud computing
ii. Public cloud computing
iii. Enhanced cloud computing
iv. Community cloud computing

These implementations are carried out by the following cloud service providers (CSP:

i. Amazon AWS
ii. Microsoft Azure
iii. Google cloud

Cloud computing is very useful to healthcare organizations, for it assists them to focus on their therapeutic services. The benefits in using CCT include (Yeng, *et al.,* 2020):

i. Easy collaboration and data sharing
ii. Mobility

iii. Cost reduction on ICT services

iv. Cost reduction in infrastructure

v. Scalability

vi. Business continuity

vii. Flexibility and

viii. Strategic values

These advantages notwithstanding, cloud computing technology has security challenges such as

i. Data loss or leakage

ii. Transaction hijacking

iii. Insecure user interfaces

iv. Denial of service attack (DOS)

v. Malicious insiders

vi. Data breaches

vii. Abuse of cloud services by CSP

viii. Estimated metering and billing system

ix. Vendor lock-in

x. Virtual machine escape

In vendor lock-in, a client can move its data and services to a CSP but cannot easily change the vendor without severe cost, legal constraints or technical incompatibility. The cyber health community in Nigeria is in its infancy, with healthcare systems and services fragmented, disjointed and heterogeneous with strong local autonomy; it is also not distributed among healthcare givers platforms (Jenyo *et al.,* 2023).

Jenyo *et al* (2023) defined an ideal cyber health system as a condition of cyber systems and networks that are not only free from malware and botnet infections, but also contribute to the overall trust and usability of the cyberspace for the wellbeing of all. This researcher and the entire research community are fighting hard to curb the menace of malware threats and botnet attacks using machine learning tools, to ensure that the cyberspace remains safe.

2.6 Cyber Threats

Cyber security threats and cyber security attacks seem to be used interchangeably. However, there are minor differences between them. A threat is a possible security violation that might exploit the vulnerability of an asset. Threats can result from accidental events, environmental, human negligence or human failure. There are four types of threats (Saeed *et al.,* 2022):

i. Unstructured threats, usually executed by inexperienced people
ii. Structured threats, which involves an organized attempt to breach a specific network or organization.
iii. External threats – These might come from individuals working outside the organization, via the internet.
iv. Internal threats – This occurs due to authorized network access. It could be due to infiltrated server account or physical access

An attack on the other hand, is a deliberate unauthorized action on a system or asset. Attacks can be active or passive, it must have a motive and should follow a preplanned method, when the need arises. Primary attack types are:

i. Reconnaissance (Probe)
ii. Denial of service (DOS)
iii. User to root (U2R)

iv. Remote to local (R2L)

While threats can be intentional or unintentional; malicious or not malicious; it can cause damage or otherwise; it can also be initiated by the system or outsiders; Attacks are deliberate, intentional, and malicious, with an objective to cause damage. It can alter or damage information, it cannot be blocked by controlled mechanism, and it is always initiated by an outsider (Saeed *et al.,* 2022).

However, in this research work, the terms threat and attacks are used interchangeably. It is an act performed by individual or computer program with harmful intent, and with a goal to steal date, cause damage or disrupt operations.

Cyber security threats can imamate from various sources such as:

i. Nation-state – These are hostile countries that can launch cyber-attack on local companies and institutions to cause disorder, interfere with their communication or inflict damage on their equipment.
ii. Terrorist organizations with ideological intent
iii. Criminal groups – to extort money from the user through phishing spamming, spyware and malware.
iv. Hackers – motivated by personal gain, revenge, financial gain and political motivation.
v. Malicious insiders – Employees, contractors and suppliers.

Electronic healthcare system (EHS) was introduced to facilitate health care delivery and health record management. It facilitates improved workflow for health care providers and increased access to patients' records, for better service delivery at lower operational cost (Alhassan *et al.,* 2016). However, EHS has become one of the attack targets by adversaries due to security vulnerabilities. These threats (attacks) reduce the privacy of patients' records and erode the confidence EHS has enjoyed.

A threat is any action or event that may lead to malfunction of the host system or exposes the patients' records to threat actors (Alhassan *et al.,* 2016), thus infringing on security attributes like confidentiality, integrity, availability, authentication, access control and non-repudiation (Tatam *et al.,* 2021; Alder, 2024). A threat could also be defined as the combined probabilities of unwanted events and their impact on the target object (Sardi *et al.,* 2020).

Threat is again defined as different kinds of likely actions that are caused by either stealth or natural means against a functional system (Idris *et al.,* ND; Tatam, *et al.,* 2021). A cyber threat actor is a person or entity that undermines security attributes (Idris *et al., ND).* To identify threats in a host system or network requires the visibility of suspicious events identifiable via indicators. An indicator of compromise (IOC) is an action perpetrated by the adversary. Advanced persistent threats (APT) are attack groups that remain undetected after compromising a host system or network (Tatam *et al.,* 2021).

Given the resistant to detection of attack groups, research community has advocated the combination of classifiers (or models) using machine learning (ML). This is achieved using either hybridization or ensemble learning methods. To model threats in EHS, one needs to identify assets of the EHS, access points, threats and rates the identified threats. Threat modeling identifies and provides visibility to threats affecting an organization (Tatam *et al.,* 2021).

An asset is any valuable component of a system that attracts the attacker, while an attacker is a person or process or program that constitutes a threat to the asset (target object). The attacker can be within the system (inside threat) or outside threat. For EHS, assets constitute the various hardware and software components and **actors** that interact with the assets. Figure 2.8 depicts actors' interaction with an asset.

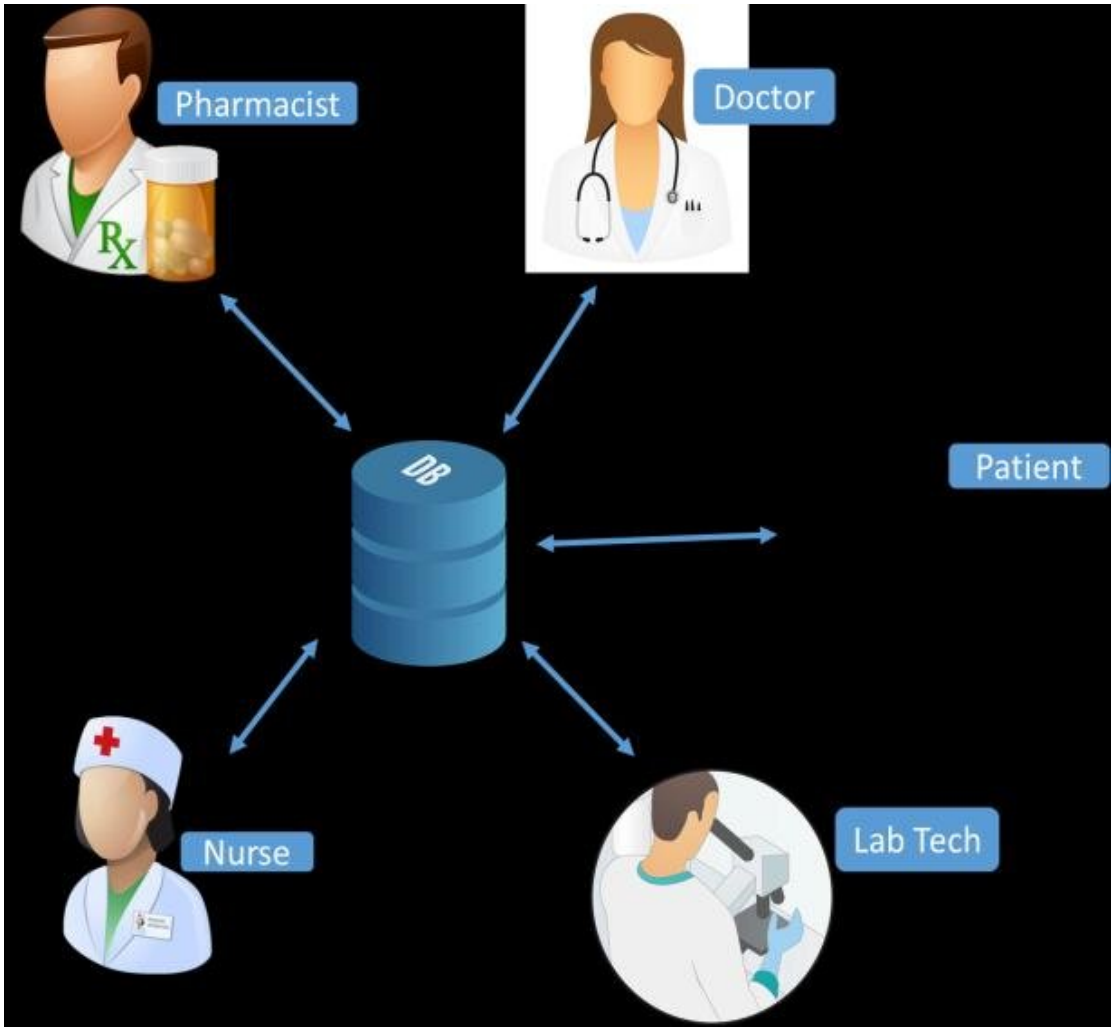


Figure 2.8: Threat actors interacting with the database (An asset)
Source: Alhassan et al. (2021)

The actors in Figure 8 include

i. The patient

ii. The Nurse who creates the patient's profile and generate identity code

iii. The physician, who records his diagnoses of the patient in HER

iv. The Laboratory scientist or technologist, who accesses the data in the EHR and conducts tests, and analysis of the samples provided by the patient and records in the database.

v. The pharmacist, who fills the prescription and provides the recommended drugs.

An access Point – these are the various interfaces used by attackers to obtain unauthorized access to the asset. Access points include:

i. Hardware ports
ii. Login screens
iii. User interfaces
iv. Open sockets
v. Configuration files.

Similarly, threats result from activities of inside attackers (who are privileged to be authenticated and authorized to access the system), and external attackers. Threats are borne from weakness in the design, coding bugs, and configuration of the hardware components. Some threats include:

i. Spoofing –This is a situation where by a person or program masquerades as legitimate through false identity to gain unauthorized access
ii. Tempering –Tempering involves changing the data to escape detection during an attack.
iii. Repudiation – This is the situation where by one of the business partners denies involvement or knowledge in the business
iv. Confidentiality – This is the situation where by information is unlawfully disclosed to unauthorized users
v. Denial of service (DOS) – this attack type bombards the target object with irrelevant requests to wear it down, and deny access to legitimate users
vi. Social Engineering –This is the act of luring the user to disclose confidential information. Some of such practices include shoulder surfing, pretexting, phishing,

man-in-the-middle (MITM) attack and code injection. Figure 2.9 depicts threat actions on the actors who access the EHR in their various duties.

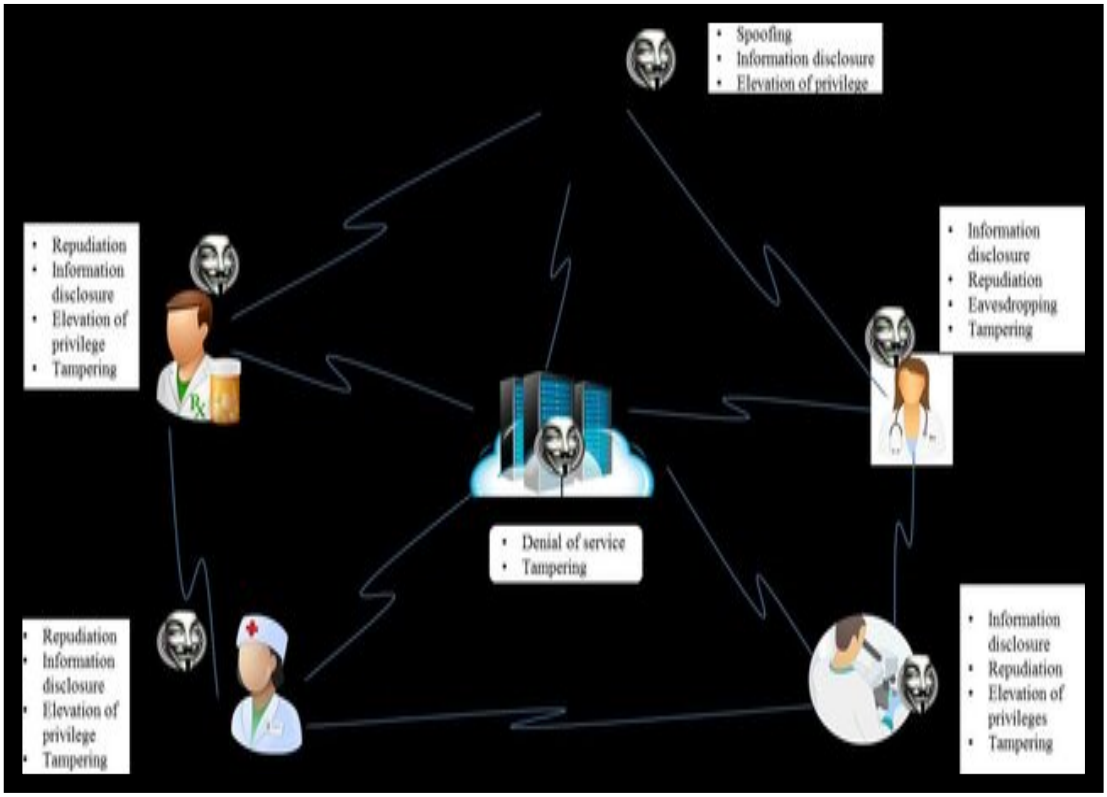


Figure 2.9: Threat Actions on the Various Actors and Asset
Source: Alhassan *et al.* (2021)

Figure 2.9 shows the database as the central asset that all users interact with by using the various interfaces available to them. Threats are ranked in terms of the damage potential, reproducibility, exploitability and discoverability. DOS is one of the attack methods, it can deprive user who connects to the EHS via the browser from accessing it. And to every actor, the related features of the threat are shown.

The most significant cyber threats include:

i. Malware, which are malicious software designed to infiltrate, damage, or gain unauthorized access to computer systems. They include viruses, Trojan horses, worms, ransomware, rootkit, key loggers, and so on.

ii. Phishing – This is a form of social engineering attack in which users are tricked to reveal sensitive information, or they install other malware by masquerading as legitimate entities.

iii. Advanced Persistent threats (APT) – These are stealth and continuous cyber-attacks, often sponsored by nation-state to target specific organizations and steal sensitive data, intellectual property, and disrupt operations.

iv. Insider threat – These are security threats originating from within the organization, either from malicious (aggrieved) insiders or negligent employees who expose systems or critical data to external threats, or contractors who are privileged to work in the system as suppliers or insurers.

v. Distributed Denial of Service (DDOS) attack – They attempt to obstruct the flow of traffic to or from a target server or asset by overwhelming it with unintelligent internet traffic from multiple sources.

Table 2.3 summarizes some of the cyber threats in recent times.

Table 2.3: Cyber incidents in recent years

Source: Katiyar *et al.* (2024)

Year	Incident	Impact
2017	WannaCry ransomware	Infected over 2000,000 across 150 countries
2018	Marriott Data breach	Exposed personal data of 500 million guests
2019	CapitalOne data breach	Compromised data of over 100 million customers and applications
2020	SolarWinds supply chain attack	Affected 18,000 customers including government agencies
2021	Microsoft Exchange Server Vulnerability	Impacted 30,000 US organizations and 250,000 organizations globally.

2.6.1 Malware Penetration Techniques

Malware penetrates computer systems including PCs, laptops and mobile devices through various ways, some of which are as follows (Arshad *et al.,* 2016; Abraham, 2017).

i. **Repackaging** – Malware developers decompile popular applications in Google play store, infuse or embed malicious contents and recompile, then distribute the impregnated (fake) application packages through third party stores. Repackaging is done through reverse engineering. Reverse engineering is a form of recovering or retrieving the source code from machine code (0s and 1s) in order to analyze the program and see what it does, subject to modification as in the case of repackaging. After embedding the code, the signature of the repackaged application is changed to enable it bypass static analysis. According to Arshad *et al.,* (2016), 77% of the top 50 free applications available in Google play store are repackaged.

ii. **Drive by download** – This is an unintentional download, which occurs when a user visits a malicious site unknowingly, and malicious code is injected into the victim's device. It could be an authorized download, without the user understanding the consequences of granting the requested permission (Arshad *et al.*, 2016; Abraham, 2017).

iii. **Homogeneity** – This is a situation whereby all the systems are running on the same platform, and connected to the same network; thus, increasing the chances of a worm in one computer spreading to other systems on the network (Arshad *et al.*, 2016; Abraham, 2017).

iv. **Dynamic payload** – dynamic payload is an encrypted malware, embedded into the device as a resource file in the system's application programming kit (APK), at run time. After installation, the malware decrypts the

encrypted payload and executes the code. Some codes are downloaded dynamically without being noticed by static intrusion detection systems (Arshad *et al.*, 2016; Abraham, 2017).

v. **Stealth malware technique** – Malware developers take advantage of the vulnerability of networked devices (host systems, laptops and mobile devices), which include low battery power, limited memory, and low processor speed; and obfuscate the malicious code to bypass the anti-malware system. Different stealth techniques used to attack victims' devices include: key permutation, dynamic loading, native code execution, code encryption, and java reflection. Vulnerability is a security defect in software and hardware that can be exploited by malware. It could be a design flaw, programming error or neglect of physical network components in the company, etc., (Arshad *et al.,* 2016; Abraham, 2017).

vi. **Zero-day vulnerability** – This is an undisclosed flaw that hackers exploit. It is called zero-day because it was not publicly reported before becoming active.

These malware penetration techniques are summarized into four types of attacks.

2.6.2 Common Malware Types

Malware comes in various forms, powered by various agents, such as (Abraham, 2017):

i. **Computer virus** – A computer virus is a contagious code or program that attaches itself to other software programs; and requires human interaction to propagate. It hides within computer executable files and modifies them in such a way that when a victim's file is executed, the virus is also executed (Yin and Song, 2019; Abraham, 2017). Such executable files have the following file extensions: .EXE, .BAT, .JS, .VB and. SCR. Viruses are the

only malware that infect other files, making them hard to be removed because they attach themselves to legitimate files; as such, most anti-virus programs rather delete the infected files or quarantine them (Abraham, 2017).

ii. **Worms** - A worm is standalone infectious software, which targets operating system files. While viruses attach themselves to existing files, worms carry themselves in their own containers. Worms usually show up through e-mails and instant messages. They use computer network to spread and do not attempt to change the system they pass through. Examples include Melissa, Morris, Mydoom, Sasser, Blaster and Mylife (Abraham, 2017).

iii. **Trojan horse**– A Trojan horse is a malicious program that masquerades as a legitimate or useful program. It spreads in the guise of routine software that persuades its victim to install it on his or her computer system. That is, it must be executed by its victim. Trojan payload is usually a backdoor malware that gives other malware like key loggers, tootkit, and more, unauthorized access to the affected computer or mobile device. Key loggers capture account names, passwords, credit card numbers, and pass all to the command and control (C & C) server. Trojan horses also steal user IP addresses, passwords, and banking details. Indeed, some Trojan horses even masquerade as anti-virus software, when actually they introduce malware to the system. Trojan horses are even developed and used by Government agencies like FBI and NSA. Examples include Magic Lantern, Finfisher, WARRIORS PRIDE (Abraham, 2017).

iv. **Hybrid and exotic forms** – Today, Trojans and Worms combine (hybridize) to form malware; and occasionally combine with a virus to form

a complex malware. The malware appears to the end user as a Trojan, but once executed, it attacks other victims over the network as worms. Examples include rootkits or stealth programs. Rootkits attempt to modify the underlying operating system and take absolute control of the system; and hide from anti-malware programs. To remove such malware, you need to remove the controlling component from memory. Bots are combination of Trojans and worms that attempt to make compromised systems (zombies) part of the larger malicious network (botnet). Botmasters are servers where zombies, the infected systems, take instructions from and send information gathered to, respectively (Abraham, 2017).

v. **Ransomware** – These are malicious programs that encrypt user data and hold them hostage waiting for a ransom to be paid in cryptocurrency before unlocking the system. They are Trojans and spread through social engineering (Yin and Song, 2019). Some examples include Reveton, Cryptolocker, Cryptowall, Wanacry, and Fusob. Fusobs affect computers and network (Abraham, 2017).

vi. **Rootkits** – This is a collection of malicious software that attempts to undermine operating system of the device and take control of the device. They usually work at the background so that users would not notice their presence. Rootkits are difficult to remove, some of which are NTRootkit and Sonny BMG Copy protection rootkit (Abraham, 2017).

vii. **Key loggers** – This malware records all the information typed using the physical keyboard. They cannot use or manipulate the virtual keyboard. The gathered information is sent to (C & C) servers (Abraham, 2017).

viii. **Grayware** – These are unwanted applications and files that though not classified as harmful, but can worsen the performance of computers and lead to security risks. Examples include Adware and Spyware (Abraham, 2017).

ix. **Adware** – Adware is used for advertising. They usually show pop-up adverts in windows that cannot be forced to close. They are nuisances but not harmful (Abraham, 2017).

x. **Spyware** - Spyware constantly spy on their victims. Their main purpose is to keep track of internet activities, including that of organizations, without the knowledge of users (Abraham, 2017).

These malware types have ways of evading detection from anti-virus programs and intrusion detection systems (IDS).

2.6.3 Malware Evasion Techniques

Several techniques exist that malware use to evade detection. These techniques are so sophisticated that anti-malware developers are struggling to catch up with.).

i. **Encryption** – Malware developers encrypt their payload to evade detection through static analysis. Encrypted malware is made up of two parts: the decryption loop, and the main body of the code. The decryption loop is capable of encrypting and decrypting the code of the malware. The main body is encrypted with either XOR or Advanced Encryption Standard (AES) algorithm. Anti-malware solutions must decrypt the main code to get to the valid signature and detect the malicious piece of software.

ii. **Oligomorphism** - The purpose of this technique is to produce a different decryption code on every infection. This technique can be detected but requires more time as the decryptions are many and randomly chosen

iii. **Polymorphism** – In polymorphism, the malicious code (payload) is modified to produce a new variant, using dead code mechanism; while maintaining its original algorithm.

iv. **Metamorphism** – This technique uses mutation engine to change the body (malicious code) on every compilation. Mutation creates slightly different versions of the same object called alleles - it generates new alleles.

v. **Obfuscation** – Hiding information by manipulating strings defeats intrusion detection systems (IDS) that are based on signature recognition. Encrypting malware is the first stage in bypassing signature-based solutions as they do not have readable information to compare. Some of the coding techniques used to evade detection through obfuscation include:

   a. *Dead code insertion* – It simply adds ineffective instructions to a program to change its appearance, but its behavior remains intact.

   b. *Register reassignment* – It simply changes or switches registers from one version to another.

   c. *Subroutine reordering* – A subroutine is reordered in a random way giving multiple variants.

   d. *Code transposition*- It reorders the sequence of the original code using unconditional branching or based on independent instructions.

   e. *Code integration*- A malware joins its code with a valid program by decompiling the original one and rebuilding it with infected code.

vi. *GPU-Assisted Malware* – Graphics Processing Units (GPU) are compressors that display 2D and 3D graphics information on the screen in real time, including videos, visual computing, and displays. It provides real time visual interaction with computed objects via graphics images and videos. GPU assisted malware does not run on virtual machines, as it uses GPU libraries. Also, GPU assisted malware has the ability to access host memory directly, making it to share CPU and GPU.

Hence, malware developers use GPU to perform tasks that might be noticed on CPU, just to go undetected (Vasiliadis, Polychronakis, and Joannidis, 2014).

Whereas malware have many ways of evading detection, they however, have many ways of leaving behind trails of their presence on a device (like smart phone).

2.7 Nigeria

In the global setting, Nigeria is a West African country and occupies an area of 923,768 square kilometers (Attah, 2017). Nigeria has an estimated population of two million people, made up of a plurality of ethnic groups with many languages. However, her official language is English. She gained political independence from Britain in October 1, 1960.

Administratively, Nigeria is divided into 36 states and a federal capital territory, Abuja. The States are further divided into 774 local government areas (LGAs). This implies that Nigeria is divided into Federal, State and Local governments.

Figure 2.10 depicts map of Nigeria showing the 36 states and federal capital territory

.

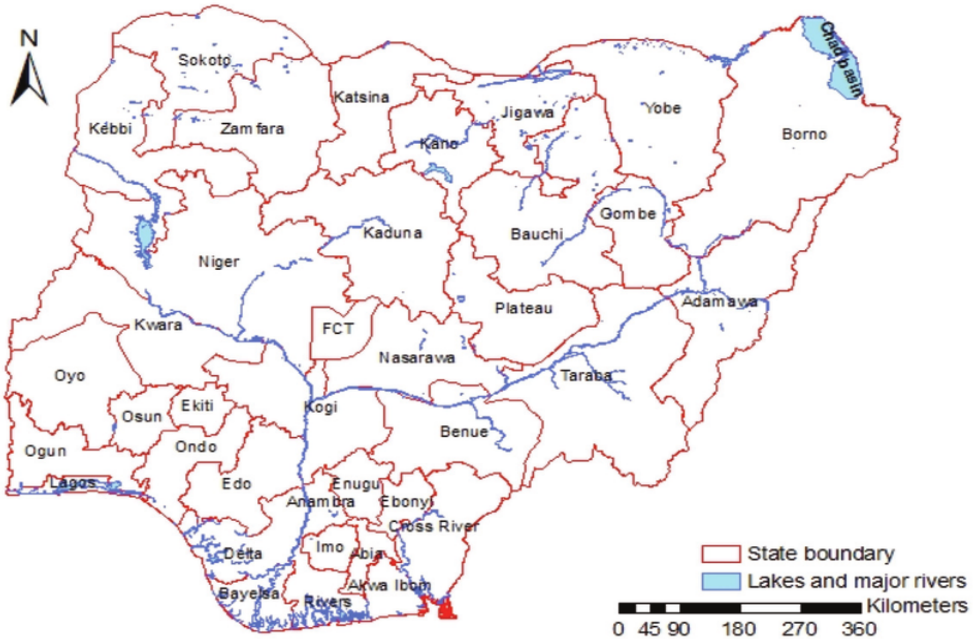


Figure 2.10: Map of Nigeria with Thirty-Six States and FCT
Source: Ishaku (2011)

2.7.1 The Nigeria Healthcare System

In Nigeria, healthcare is subdivided into three tiers

i. The traditional healthcare system

ii. Public healthcare system

iii. Private healthcare system

The public healthcare system is further divided into

i. Tertiary healthcare

ii. Secondary healthcare

iii. Primary healthcare systems

These orthodox healthcare systems are managed by the three tiers of government respectively. Figure 2.11 depicts the Nigerian healthcare system.

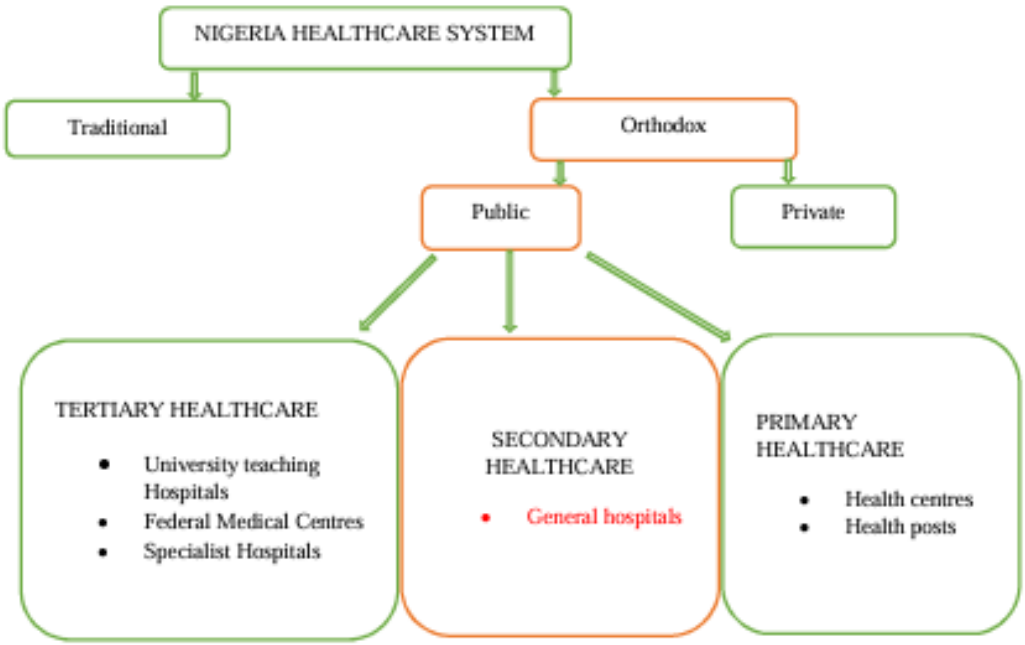


Figure 2.11: Healthcare Delivery System in Nigeria
Source: Attah (2017)

From Figure 2.11, the tertiary healthcare systems are managed by the federal government, and the secondary healthcare system is managed by the state governments; while the primary healthcare systems are managed by the local governments. While the organization of the healthcare systems seems well coordinated, the practical working of the setup does not as plausible. There are duplications of responsibilities and confused roles among the three tires of government, corruption in the system or absence of basic infrastructure. No drugs in the hospitals, no equipment, no beds, and in the primary level, there are no doctors nor nurses.

However, the private sector seems better organized than the government sector, with better facilities, personnel and infrastructure, but costly and near beyond the reach of the common man.

Traditional healthcare systems are seemingly not recognized by the government of Nigeria, rather, emphasis is on orthodox medicine. Nigerian patients, who seem to know what sickness is for orthodox medicine and which is for traditional care, patronize traditional medical care at their own discretion and risk. In developed countries, in particular Asian countries like China, traditional medicine is carried along with orthodox medicine and co-funded by government and private companies or individuals.

2.7.2 Electronic Health Systems in Nigeria

Efforts to develop e-Health in Nigeria started in 1994, but these efforts are piecemeal, uncoordinated and at experimental (pilot) stages (Attah, 2017). Barriers to e-Health implementation in Nigeria include:

i. Lack of national e-health strategy and policy
ii. Lack of e-health legislative framework
iii. Epileptic electricity supply
iv. Poor communication networks and facilities, with very loosely controlled mechanism by Nigerian Communications Commission (NCC).
v. Little or no ICT personnel and infrastructure in the healthcare systems
vi. Inadequate government commitment in terms of policy, legislation, funding and more.
vii. Malicious attacks on the pilot systems by malware and botnets, amidst internal and external aggressions

It is in the light of these problems that the researcher is motivated to join the research community to design and develop a framework that will use machine learning tools to detect the malicious software (malware threats) and suggest remedial option

2.8 Soft Computing

Cyber security is seriously being undermined by threats and this affects individuals, organizations, and society at large. Malicious actors (malware developers and hackers and even programs) and constantly developing variants of known malware signature families to compromise computer networks, host systems and intimidate innocuous users; the threats also steal sensitive or critical data and disrupt operations and services (Katiyer *et al.,* 2024). Cyber security approaches based on signature-family detection, and manually defined security policies could not keep pace with the ever-evolving threats landscape. Hence the use of machine learning tools to solve such practical problems.

Soft computing also called computational intelligence, is an aspect of artificial intelligence (AI) that trains computers or computer programs to perform the tasks of experts in specific fields. Machine learning (ML) is a subset of soft computing, which teaches computer to learn and improve on past experience without being programmed. With ML techniques, cyber security systems can analyze huge data to uncover hidden patterns, detect subtle anomalies and make intelligent decisions to prevent, detect and respond to cyber intrusions. ML algorithms have unparalleled capacity to analyze network traffic, user behavior and system logs in real-time.

Cyber security threats are analyzed and detected by three approaches:

i. Misuse or static approaches
ii. Anomaly or dynamic approaches
iii. Hybridized approach

Figure 2.12 depicts the three machine learning techniques that are used to detect malware intrusions.

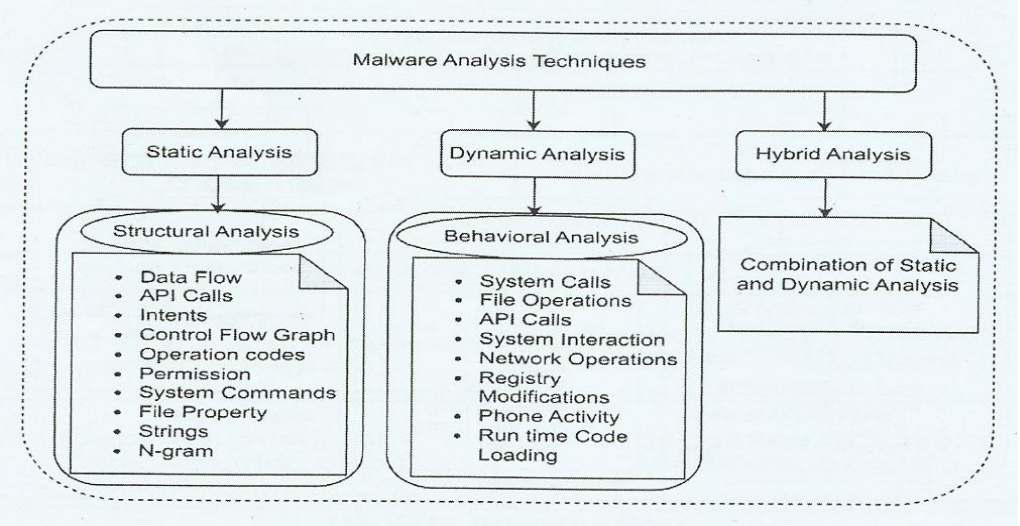


Figure 2.12: Taxonomy of Malware Analysis Techniques
Source: Smmarwar *et al.* (2024)

In misuse type, reliance is on signature-based detection, where known threat patterns are identified and blocked based on predefined rules and databases of signature families (Black lists). These techniques are very effective in detecting known threats and speedily too; they however, cannot detect unknown (new) malware and zero-day threats. More so, maintaining these families requires constant updating and can lead to false positive rates (FPR). Static analysis uses binary codes to identify harmful files and viruses. The biggest drawback of static analysis is the absence of program source code, and it is difficult to extract binary code (Kumar and Das, 2023).

Anomaly based detection aims at identifying deviations from normal applications or user behavior. However, these techniques are prone to high false positive rates and can be difficult to tune and maintain over time. More so, these anomaly detection types often operate in silos, focusing on specific aspects of the network or system such as endpoints, servers or applications, instead of focusing on the entire system or network. This fragmented approach can lead to blind spots and inefficiencies in detecting and responding to threats spanning multiple dimensions (Katiyar *et al.,* 2024).

Hybridized approach combines both static and anomaly approaches to produce a stronger and more effective intrusion detection system, to foil the sophistication of malware. Table 2.4 depicts ML malware detection techniques or methods, highlighting their features, advantages and disadvantages.

Table 2.4: Comparison of ML Based Malware Detection Approaches

IDS Techniques	Features	Advantages	Limitations
Static Analysis	File properties, byte sequences, and header information	Fast, low resource requirements, and comparatively low FPRs	Can be evaded using obfuscation techniques, and packing techniques.
Dynamic Analysis	API calls, Network traffic system or resource usage.	Captures run-time behavior, resilient to obfuscation attacks	Higher resources consumption, potential sandbox evasion.
Hybrid Analysis	Combination of both static and dynamic features	Improved accuracy, robustness to evasion techniques.	Increased complexity, may require manual feature engineering.

Source: Kaliyar *et al.* (2024)

2.8.1 Intrusion Model

Intrusion is an attempt by unauthorized persons or programs to undermine the security attributes of data: confidentiality, integrity and service availability (CIA) by exploiting their vulnerabilities (Bhuyan, Bhattacharyya, and Kalita, 2014). This attempt is perpetrated by malicious programs (malware) written and deployed by malware developers to achieve their nefarious intents. The diamond thread model is used to apply scientific principles to intrusion analysis. It maps the relationships and capabilities of adversaries to the target object. It consists of four main parts, as depicted in Figure 2.13, of an intrusion activity: the adversary using his or her capabilities to a target's infrastructure. It tracks attack groups because they usually change their targets, and their links over infrastructure against a target.

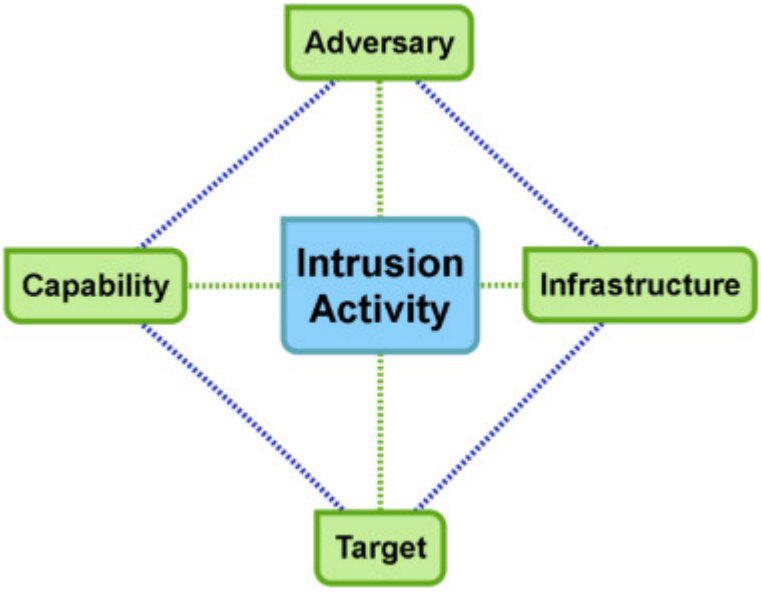


Figure 2.13: The Relationship Between the Attacker, His/Her Capabilities, and the Infrastructure of the Target Object
Source: Tatam *et al.* (2021)

The Adversary:

An intruder must have a reason for deciding to breach the security of a network. The reason, for modern malware developers, is to make money.

Capabilities:

An intruder must have the ability to carry out an intrusion, such as programming knowledge or access to attack gadgets, most of which are freely available on the net.

Infrastructure:

An intruder must have access to the network or the organization's infrastructure either through flaws in the security plan, bugs in software program, or physical proximity to network components.

The Target:

The target is the organization, the host system or network belonging to the organization, with the intent of disrupting the operation of the target.

Out of these four features, security experts or user or system Administrator can only influence or control the infrastructure; for capabilities dependent on the personality of the intruder or the type of data built into the network; there is no way anybody can stop a would-be hacker from acquiring programming knowledge or from obtaining the tools necessary to use in his or her nefarious intent (Ibrahim *et al.,* 2017). It is the responsibility of system Administrator to ensure that the network infrastructure is properly secured.

A good network security system should conceal all open ports, exploitable applications, and indeed, formulate good access control mechanism (ACM); the system should also be easy to use. These characteristics are achieved through the use of firewalls, cryptography, steganography, anti-virus applications, intrusion detection systems (IDS), and intrusion response systems (IRS) (Ibrahim *et al.,* 2017). Intrusion detection system (IDS) is one of the second-degree security mechanisms that detects and reports malware attacks.

2.8.2 Intrusion Detection Systems (IDS)

Intrusion is a set of activities used by adversaries to compromise systems or network components in terms of confidentiality, integrity and availability. It is a deliberate and unauthorized attempt to access information, manipulate data, and render a system unreliable or unusable (Bhuyan *et al*, 2014). These threats are achieved by either an insider of the system or outside agent, to gain unauthorized access and control of security mechanisms, if protective measures are not put in place.

To protect these security attributes (CIA), intrusion detection systems (IDS) are used. IDS gathers and analyzes information from various areas, within a host or a network, to identify possible security breaches, and report either to the user or system Administrator (Bhuyan *et al,* 2014). IDSs are broadly categorized into static or misuse-

based detection systems; and dynamic or anomaly-based detection systems. Misuse or static analysis techniques decompose or dismantle or de-compile applications off-line, and analyze their features and characteristics against signatures of malware families already gathered into a database. They are very efficient and fast in their analysis and classification between malicious and benign applications, with minimal false alarm (FPR) (Baby and Jeba, 2017; Azad and Jha, 2014; Dewa and Maglaras, 2016; Kruegel, N.D).

However, there are drawbacks to static analysis techniques: they are easily deceived with slight variations of the signatures they know; they cannot detect zero-day attack, nor can they detect new and unknown attacks. Signature variations are carried out using obfuscation techniques such as polymorphism and metamorphism (Baby and Jeba, 2017; Azad and Jha, 2014; Dewa and Maglaras, 2016; Kruegel, N.D). Polymorphic obfuscation encrypts the payload to evade detection, only to decrypt it back during execution. This is achieved using the following methods:

i. no operation (NOP),
ii. dead code insertion,
iii. code transposition (changing the order of instructions, and
iv. placing jump instructions, to maintain the original meaning (semantics),
v. register reassignment,
vi. behavioral insertion.

Whereas polymorphism attempts to encrypt only the payload, metamorphism encrypts the entire malicious application. These evasions are possible, partly because emphasis is placed on grammatical structure (syntax) of malware algorithm, ignoring the meaning (semantic) assigned to the symbols, characters, and words used in the grammatical

structure of the malware algorithm (Baby and Jeba, 2017). Grammar is a set of rules that defines whether or not the sentence is properly constructed.

However, security industries, as at now, are using static analysis method, which is pattern matching of targeted applications with family signatures found in the database. The consequences being regular update of the databases, inability to identify unknown malware and zero-day attacks, inability to detect dynamic code loading, native code reflection, java reflection and encrypted code. (Wang *et al,* 2015), Inability to analyze the source code (Tatam *et al.,* 2021). Figure 2.14 illustrates misuse-based IDS,

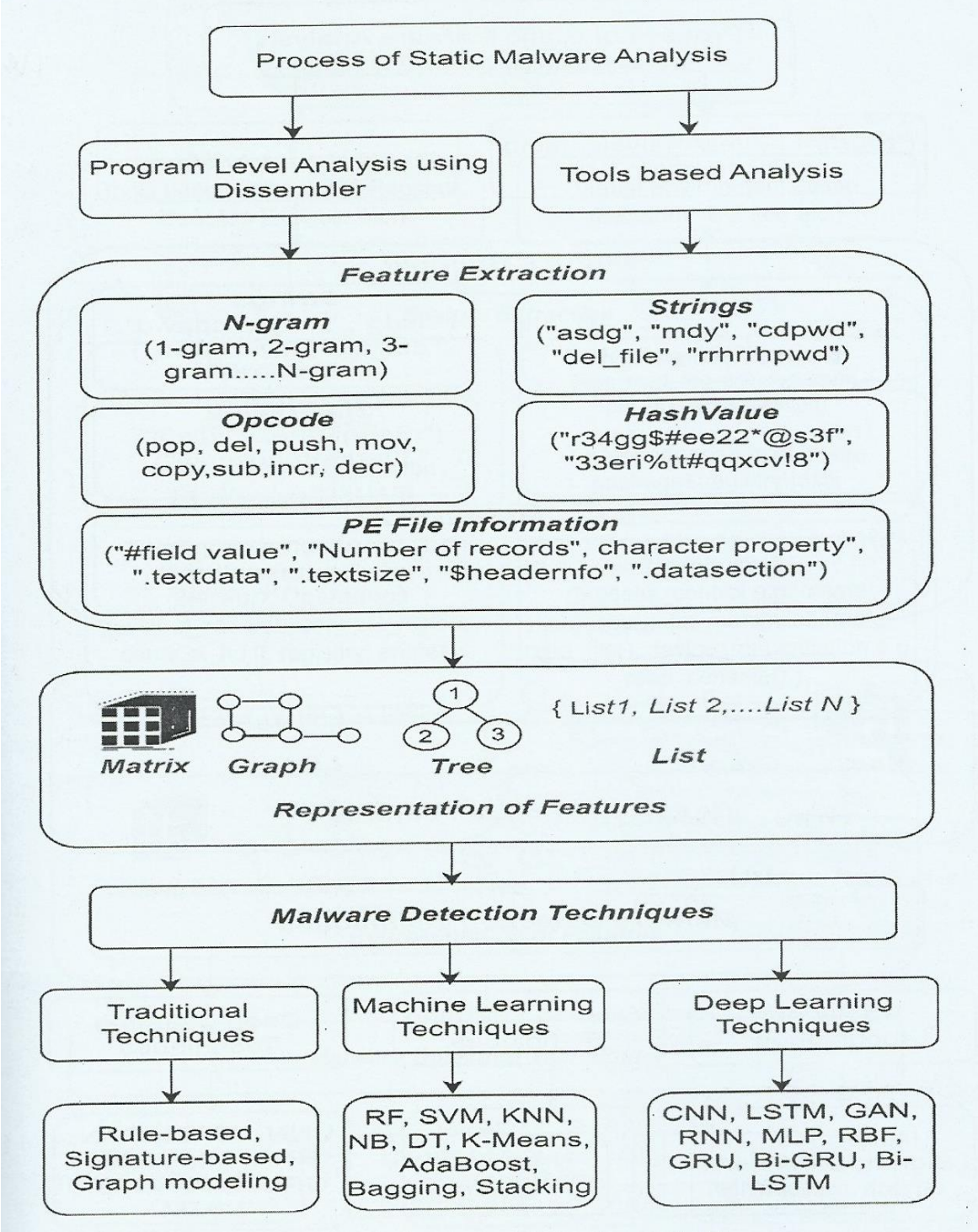


Figure 2.14: Schematic Diagram of Static Analysis for Malware Detection
Source: Smmarwar et al. (2024)

Anomaly based detection techniques are based on building and training classifiers (models) on normal behaviors, which is used as a basis to observe deviations by anomalous applications (Baby and Jeba, 2017). Dynamic IDS is trained to learn the

behavior and characteristics of normal applications, and report any deviation from the norm as malware. Some of the characteristics of normal applications include:

i. In Attribute Relation File Format (ARFF) file of NSL-KDD dataset, normal applications use TCP and UDP protocol types; while malware uses ICMP protocol type to execute DDOS attack on target objects such as ICMP flood, smurf attack with spoofed or faked IP address, and ping_of_death attack to cause buffer overflow and possible crashing of the device. Also, normal applications use SF flag, while malware have affinity for S0 and REJ flags.

ii. POLP – Principle of least privileges (Lord, 2020) states that "persons, applications, or processes be given the barest minimum resources to complete their tasks". Normal applications obey POLP principle by requesting for permissions and other resources within the acceptable range, but malware request for more than necessary resources for their nefarious activities.

iii. Android install time permissions – Permissions form the first barrier to malware attack. Normal applications declare permissions from manifest.xml file in keeping with POLP principle; but malware in addition to those in the manifest.xml file, ask for permissions to access hardware resources like sensors, camera, GPS, GPRS, and undocumented APIs from third party stores.

iv. Central processing unit (CPU) – Normal applications use CPU to process their data; but malware tends to use graphics processing unit (GPU) to process data and evade detection.

v. Dangerous permissions – These permissions tend to violate user privacy by requesting to access contact lists, photo albums, user ID/password,

credit/debit card numbers, etc. Malicious applications tend to use these permissions in their manifest.xml files, while normal applications request for normal permissions that do not temper with user privacy.

These techniques have two major advantages over misuse techniques: they are able to detect unknown attacks, as well as zero-day attacks; the aforementioned learned normal behavior is customized for every system, application or network. This makes it difficult for attackers to determine the attack pattern to use without being detected. The drawbacks of these techniques include: high resource consumption, high percentage of false alarms and difficulty in determining the event that triggers alarm; inability to detect privileged permissions, emulation detection system is prone to evasion by malware, and subject to low code coverage (Wang *et al.*, 2015). On the other hand, most researches in dynamic analysis use one subset of malware features to represent its behavior pattern and ignore other ones. For instance, API based sequences are used in analysis, ignoring network-based sequences which use Packet Capture (PCAP), an application programming interface for capturing network traffic, and files to extract the network flow information (Mashiri *et al.*, 2017). Also, it is observed that anti-virus vendors label malware samples differently, making them inconsistent with each other. Therefore, labeling malware in dynamic analysis may be less accurate, given the operators' approach in their heterogeneous methods (Mashiri *e al.,* 2017). These drawbacks have made the implementation of anomaly detection techniques slow and difficult to adopt. Figure 2.15 illustrates dynamic or anomaly intrusion detection system.

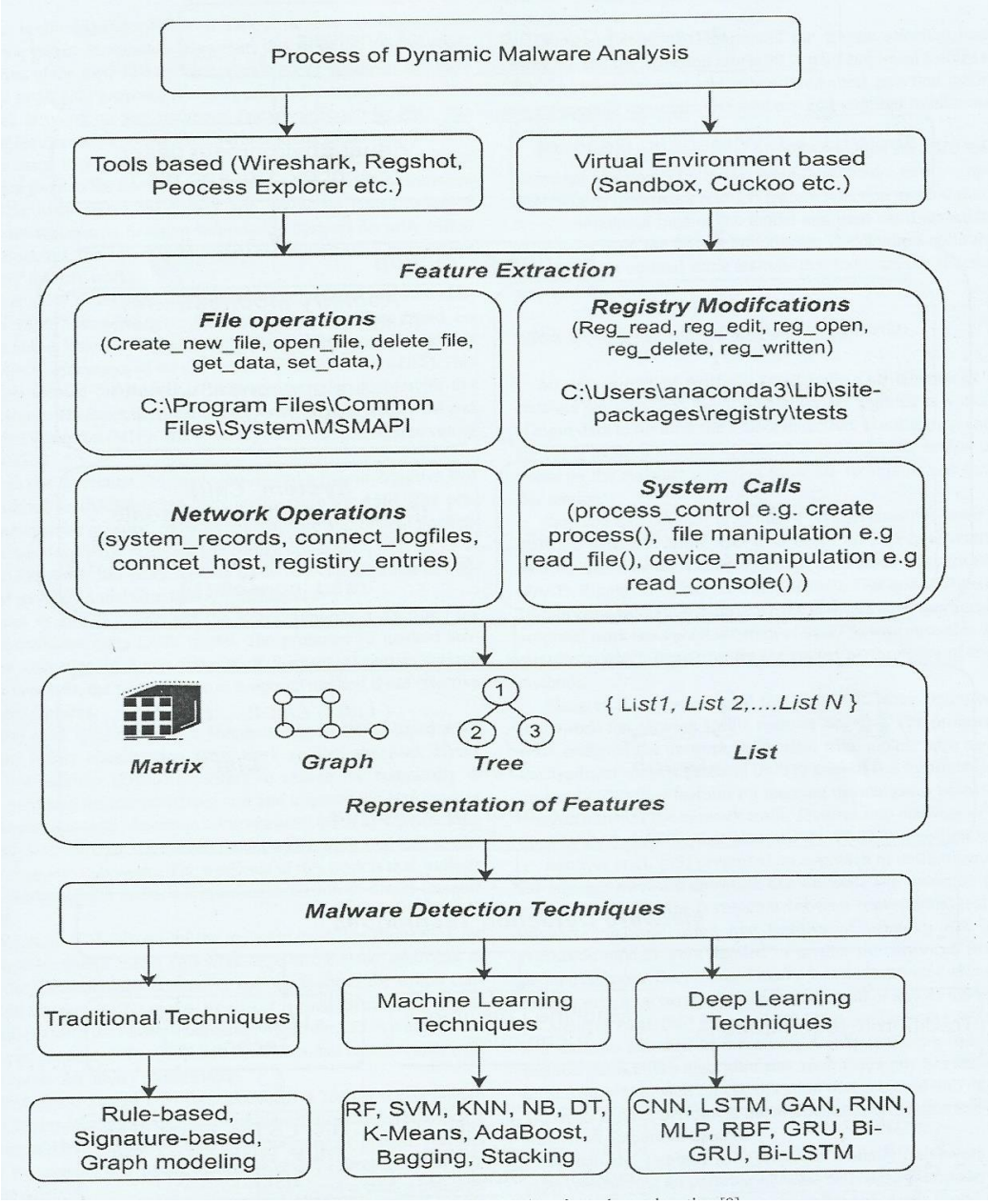


Figure 2.15: Schematic Diagram for Anomaly Detection of Malware
Source: Smmarwar *et al.* (2024)

From the foregoing, none of the aforementioned techniques is unique; they all have their strengths and weaknesses; as pointed out in "no free lunch" theory (Baby and Jeba, 2017, Bui *et al.,* 2017). The limitations of traditional techniques (misuse and anomaly types), coupled with increasing volumes, velocity and variety of threats has necessitated the use of machine learning techniques to solve the problems of incessant threats

(Kumar and Das, 2023). Using ML techniques to fight cyber threats has the following advantages:

i. Improved threat detection – ML algorithms can analyze massive datasets to identify patterns and anomalies that may indicate malicious activity, enabling the detection of previously unknown and zero-day threats.
ii. Faster incident response – ML models can automatically triage and prioritize security alerts, this reducing delay in response to attack initiated manually.
iii. Adaptive and scalable protection – ML models can continuously learn and adapt to new threats providing a more flexible and scalable approaches to cyber security compared to rule-based systems.
iv. Predictive analysis – By analyzing historical data and trends soft computing (SC) techniques can help predict potential future threats and vulnerabilities.

Some of these ML algorithms include Artificial Neural Network (ANN), Fuzzy logic, evolutionary computation (genetic algorithm and genetic programming), probabilistic computing including (KNN, SVM, NB, DT, RF, LR, DL (Deep learning)), and so on.

Artificial Neural Network (ANN) – This classifier predicts the behaviors of various users and daemons in the system. Its major advantage is it infer solutions from data without prior knowledge of the data structure. It is also tolerant to imprecise data and the uncertainty in it.

Fuzzy system – This algorithm or classifier models uncertainty in human expressions (experts) using Triangular Fuzzy Numbers (TFN) to convert the expressed variables or phrases into crisp values

Decision Trees – Decision trees are used to reduce variance in data sets. Its graphical representation consists of

i. Root – initial node

ii. Nodes – feature attributes

iii. Arcs – labels' categorical values

iv. Leaves – category of classes

Random Forest – This is a mega algorithm in ensemble bagging that combines the output of multiple decision trees, and other models to produce a more robust model or classifier. It does not require scaling or normalization. It can be used for classification or regression.

Naïve Bayes- This is a probabilistic classification based on Bayes' theorem, which assumes that the features are conditionally independent given the class label.

K-Nearest Neighbor – This is a non-parametric method that classifies new instances based on the majority class of the k nearest training instances in the feature space.

Support Vector Machine – This is a supervised learning classifier used for classification problems. Although it can be used in linear level analysis, it is most appropriate in higher level feature space. Basically, SVM is used for binary classification, it can also be used for multi-class problems.

Logistic Regression – Logistic regression is a classification algorithm. It is a data analysis technique that uses statistical models to find the relationship between two data factors. It then uses this relationship to find or predict the value of one factor based on the others. The prediction usually has a finite value say Yes or no.

The need to combine machine learning (ML) techniques based on their complementing features is now (Wang and Wang, 2015). The combination will be by using ensemble learning methods or hybridization method. In ensemble learning methods, weak or base learners are trained either sequentially or in parallel. The predictions of the trained models are aggregated using a voting technique, to produce a much better and stronger

model. Hybridization involves the use of binary models, which are trained, predict their results, and combine them to produce a better result than the individual algorithms. In this research work, ensemble technique used is bagging, with Random Forest as the training algorithm; while KNN, SVM, DT, and NB are base learners.

The performance of these models depends on the quality of input data used to train them. The process of transforming the raw data into informative features is called feature engineering. Feature engineering includes the following processes

i. Extraction of statistical properties
ii. Packet header information
iii. Byte sequences from network traffic data
iv. System logs metrics.

Feature selection is used to identify relevant and discriminative features from a larger set of data. It is aimed at improving model performance, reduce over fitting and enhance interpretability. Feature selection techniques include:

i. Filter methods
ii. Wrapper methods
iii. Embedded methods.

2.8.3 Classes of Machine Learning

Machine learning techniques are grouped into three main types:

i. Supervised learning methods
ii. Unsupervised learning methods
iii. Reinforcement learning methods

Supervised learning method – In this method, the algorithm learns from labeled training data set, where the desired output is known in advance. The goal is to learn a function

that maps input features to output labels enabling the prediction of labels to new, unseen data. That is, data that it did not train with (test dataset).

Unsupervised earning methods – In this technique, algorithms learn from unlabeled data, aiming to discover hidden patterns without any predefined output.

Re-enforcement learning methods – In this type, algorithms interact with its environment and receives reward or punishment for its action. The goal is to learn a policy that maximizes the cumulative rewords over time (Katiyar *et al.,* 2024). Table 2.5 depicts the main characteristics of ML types and applications.

Table 2.5: Characteristics and Applications of ML Types

ML Types	Characteristics	Application in cyber security setting
Supervised Learning	Learns from labeled data to predict output	Malware classification and spam detection
Unsupervised learning	Discovers patterns in unlabeled data	Anomaly detection, clustering, and so on.
Re-enforcement learning	Learns through interaction with its environment.	Adaptive Network security policies, agent-based systems

Source: Katiyar *et al.* (2024)

Similarly, Table 2.6 compares the ML types and notes their strengths and weaknesses

Table 2.6: ML Types and their Strengths and Weaknesses

Techniques	**Strengths**	**Weaknesses**
Re-enforcement learning Justine (2018)	Solves complex problems It may correct errors occurring during training It involves training data obtained through interaction with the environment It is flexible and can combine with other techniques	i. It is not good at solving simplified problems ii. It requires excessive data and computation iii. It is dependent on the reward function's quality iv. It is difficult to debug and interpret.
Unsupervised learning Aukur (2018)	i. Does not require manual data preparation ii. Capable of finding previously unknown patterns in data	v. Producing inconsistent results vi. Requires post-processing or interpretation to assign labels.

Supervised learning Mohamed (2017)	iii. iv. v. vi.	It can be normalized to avoid over fitting Linear models can be updated easily with new data It produces far more accurate results, and it is more reliable Efficient in finding solutions to linear and non-linear problems.	vii. viii. ix. x.	Takes a long time to train Limited performance especially in non-linear relationships It is not cost efficient with scalable data The accuracy is dicey. A higher accuracy does not imply higher performance.

Source: Katiya (2024)

2.9 Soft Computing Algorithms

Soft computing algorithms are algorithms trained and used to solve difficult and complex problems. They include:

2.9.1 K-Nearest Neighbor (K-NN) (Fix and Hodges, 1951)

K-Nearest Neighbor (K-NN) algorithm is one of the simple, efficient and interpretable algorithms that can be applied in classification as well as regression problems. Although simple, its performance competes favorably with complex classifiers like Support Vector Machines (SVM) and Artificial Neural Networks (ANN). In fact, its performance is used as a benchmark for complex classifiers (Prasath *et al.*, 2017).

K-NN was proposed by Fix and Hodges (1951), and ranks among the top ten (10) algorithms in Machine Learning (ML). It is a non-parametric and lazy learning algorithm. Non-parametric means that there are no parameters or a fixed number of parameters, irrespective of the size of the dataset. Parameters are rather determined by the size of the training dataset; K-NN is lazy learning because it stores the entire training data in memory and waits until the test data is introduced for analysis in real-time, without having to create a learning model (Prasath *et al.,* 2017). It is broadly divided into two sub types:

i. Structure less nearest neighbor (NN)

ii. Structure based nearest neighbor (NN)

In structure less K-NN technique, the whole data is classified into training and test data samples. Distance is measured from the training points to the test sample point, and the point with the shortest distance is called the nearest neighbor (Prasath *et al.*, 2017; Salvador-Meneses *et al.*, 2019). Structureless K-NN is used in combination with categorical (nominal or ordinal) data type, which is compressible in order to reduce the memory overhead; and decompresses in real-time during classification (Salvador-Meneses *et al.*, 2019). More so, traditional algorithms have problem working with datasets having very large attributes (or features or dimensions) because such datasets consume memory space (since the entire training dataset needs to be stored in memory prior to processing). For example, NSL-KDD dataset alone has forty-two (42) features of numerical and categorical types with two (binary) classes – benign and malicious classes. In machine learning, there are several types of data, such as numerical data, categorical data, text data, images, audio, video, etc. Current techniques of Machine Learning (ML) usually convert other forms of data into numerical data types before processing.

Structure based Nearest Neighbor (NN) techniques are of type tree such as ball tree, k-d (k-dimensional) tree, Orthogonal Structure Tree (OST), fixed axis tree, nearest feature line and central line. They all rely on continuous attributes. The difference between discrete and continuous data is that discrete data is countable while continuous data is measurable. Discrete data contains distinct or separate values. On the other hand, continuous data includes any value within a range, say [0, 1].

K-NN is applied in solving many practical problems like pattern recognition, text categorization, ranking models, object recognition and event recognition (Prasath *et al.*,

2017). Its simplicity, non-parametric, lazy learning (real-time analysis) and similarity of nearest neighbors categorized it among the top ten algorithms in ML (Salvador-Meneses *et al.*, 2019).

2.9.1.1 Principle of Similarity in K-NN

K-nearest neighbor algorithm is based on the principle that "similar things exist closer to one another or like things are near to each other" (Cheng *et al.*, 2014; Salvador-Meneses *et al.*, 2019). This principle enables the use of distance measuring techniques (metrics) for K-NN in classifying test sample points using the entire training dataset in memory.

2.9.1.2 Similarity and Metrics

The focal point of K-NN is its dependence on the distance measure (or similarity measure) between the test data point and the training data points stored in memory (Prasath *et al.*, 2017). There are up to fifty-four (54) distance measuring techniques in literature, grouped into eight families: L_p Minkowski distance measures, L_1 Distance measures, Inner product distance measures, Squared Chord distance measures, Squared L_2 distance measures, Shannon entropy distance measures, Vicissitude distance measures, and other distance measures (Prasath *et al.*, 2017).

Some of the distance measures, as given by Prasath *et al.*, (2017) include:

Minkowski distance measure, Manhattan distance measure, Euclidean distance measure, Chebyshev distance measure, Mahalanobis distance measure, Cosine distance measure, Hamming distance measure, Correlation distance measure, Soergel distance measure, and Lagrange distance measure. These distance measures must satisfy certain properties before they can be regarded as metrics or distance measuring functions.

2.9.1.3 Properties of Metrics

Distance is a numerical description of how far apart entities are. The distance function between two vectors **x** and **y,** denoted as d(**x, y**), is the distance between both vectors, which should be non-negative real number. This distance function is considered a metric if it satisfies the following properties

i. **Non-negativity**: The distance between **x** and **y** is equal to or greater than zero, as illustrated in Equation 2.1

$$d(x, y) \geq 0 \qquad \text{Equation 2.1}$$

ii. **Identity of discernible**: The distance between **x** and **y** is equal to zero if and only if (iff) **x** is equal to **y**, as defined in Equation 2.2.

$$d(\mathbf{x}, \mathbf{y}) = 0, \quad \text{iff } \mathbf{x} = \mathbf{y} \qquad \text{Equation 2.2}$$

iii. **Symmetry**: The distance between **x** and **y** is equal to the distance between **y** and **x**, as shown in Equation 2.3.

$$d(\mathbf{x}, \mathbf{y}) = = d(\mathbf{y}, \mathbf{x}) \qquad \text{Equation 2.3}$$

iv. **Triangle inequality**: The triangle inequality states that the shortest distance between any two points is a straight line**.** Given the presence of a third point, z, the distance between x and z is less than or equal to the sum of the distance between x and y and y and z, as illustrated in Equation 2.4.

$$d(x, z) <= d(x, y) + d(y, z) \qquad \text{Equation 2.4}$$

(Prasath *et al.*, 2017; Chumachenko, 2017).

When the distance is in a range, say, [0, 1], the calculation of the corresponding similarity measure**,** **s**, is defined in Equation 2.5.

$$s(x, y) = 1 - d(x, y) \qquad \text{Equation 2.5}$$

Performance of KNN also depends significantly on the distance measure used. In addition, KNN is tolerant to noise, with a performance reduction of only 20% when up to 90% noise is infused into both training and test datasets (Prasath *et al.*, 2017). Within the eight families of distance measure, there are many members in each family, as illustrated in the Minkowski family.

2.9.1.4 Minkowski Distance Measures

Minkowski distance measure family consists of three distinct members

i. Manhattan measure

ii. Euclidean measure

iii. Chebyshev measure

These measures are derived from the variation of p value in Minkowski formula, as illustrated in Equation 2.6.

$$D_{Mink}(x, y) = \sqrt[p]{\sum_{i=1}^{n} |xi - yi|^p} \qquad \text{Equation 2.6}$$

The distance measures above are derived from Equation 2.14, when the p value changes: when p=1, the Minkowski formula becomes Manhattan formula, but when p=2, the same Equation 2.14 becomes Euclidean formula; and when p=∞, it becomes Chebyshev formula. Xi is the i-th value in the vector **x,** and y_i is the i-th value in the vector **y**.

2.9.1.5 Manhattan distance measure

This measure is also known as the L_1 norm, and it represents the sum of the absolute difference between the opposite values in the vectors **x** and **y.** This difference is illustrated in Equation 2.7.

$$D_{Manh}(x, y) = \sum_{i=1}^{n} |xi - yi| \qquad \text{Equation 2.7}$$

2.9.1.6 Euclidean distance measure

This distance measure is also known as L_2 norm; and it is an extension of Pythagoras theorem as shown in Equation 2.8.

$$D_{Eucl}(x, y) = \sqrt{((x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2)}$$

$$= \sqrt{\sum_{i=1}^{n} |xi - yi|^2} \qquad \text{Equation 2.8}$$

(Zhang *et al.*, 2016; Prasath *et al,* 2017)

2.9.1.7 Chebyshev distance measure

This distance measure is also known as maximum value distance, Lagrange and Chessboard distance; and it is used when two objects are defined in different dimensions. It is illustrated in Equation 2.9.

$$D_{cheb}(x, y) = \max |xi - yi| \qquad \text{Equation 2.9}$$

2.9.1.8 L_1 Distance Measure Family

Has as an example the Lorentzian distance measure, which is measured by natural logarithm (ln) of the absolute difference between two vectors. From Equation 2.10 we see that one (1) is added to it to avoid negative values or log of zero.

$$D_{Lore}(x, y) = \sum_{i=1}^{n} ln(1 + |xi - yi|) \qquad \text{Equation 2.10}$$

Distance functions or metrics use categorical data or numerical data to determine the classes in a training dataset.

2.9.1.9 Categorical Data Type

In machine learning, various types of data are used to analyze malware effects on computers, which are categorized into independent, dependent and controlled variables. Depending on the type of data, in supervised learning, a set of known variables is labeled for instance, as benign or malicious. Current techniques in machine learning usually convert other forms of data to numerical data before preprocessing. These class labels are categorized by distance functions. Hamming distance measure is used for

categorical data; while Minkowski family of distance measures is used to categorize numerical data type.

Categorical variables are values of names or labels such as red, green, blue (for color); race, sex, age group, educational levels (Salvador-Meneses *et al.*, 2019). There are three types of categorical variables: nominal, ordinal and binary variables. A nominal variable has no intrinsic ordering, for example, gender with two categories (male and female). An ordinal variable has clear ordering, for example, educational level may be categorized into primary education (1 to 6 years), junior secondary (1 to 3 years), and senior secondary (1 to 3 years). Variables can be independent, dependent or controlled variables.

Continuous variables are numerical variables that have an infinite number of values within a range [0, 1]. Discrete variables are numeric variables that have countable number of variables between any two variables. For categorical values, hamming distance measure is used to compute the distance between two variables or points. Hamming distance is the number of mismatches between two equal strings in same position. XOR operator can also be used to compute the hamming distance. This computation is illustrated in examples that follow. The hamming distance measure is given in Equation 2.11.

$$D_{ham}(x_i, y_i) = \begin{cases} 1, & \text{If } x_i \neq y_i \\ 0, & \text{If } x_i = y_i \end{cases} \qquad \text{Equation 2.11}$$

The computation of Hamming distance is illu ıle 7 with many examples.

Example 1:

Table 2.7: Hamming Distance

X	Y	Distance
Male	Male	0
Male	Female	1

Example2:

W o r k $D_{ham} = 2$

W a l k

Example 3:

L a n e $D_{ham} = 2$

V a l e

Example 4: Using binary digits

0 1 1 0 0 1 0 0 $D_{ham} = 3$

0 0 1 0 0 0 0 1

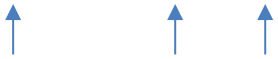


Table 2.8 illustrates the computation of hamming distance using the XOR operator

Table 2.8: Using XOR Operator to Compute Hamming Distance

INPUT		OUTPUT	
X	Y		
0	0	0	$D_{ham} = 2$
0	1	1	
1	0	1	
1	1	0	

As can be seen, with the XOR operator, when the inputs are the same, the output is zero, and vice versa. When the i-th attribute has numerical values, the range normalization difference distance is used, as illustrated in Equation 2.12

$$\mathbf{D}_N(xi - yi) = \frac{|xi - yi|}{\max(\{xi\}) - \min(\{yi\})} \quad \text{Equation 2.12}$$

Euclidean distance measure is most commonly used in conjunction with categorical data to determine classes of training data sets. However, the choice of these metrics (distance measures), given the type of data used, is influenced by the value (number) of nearest neighbors, **k**, in the training data set to the test data being analyzed.

2.9.1.10 Choosing the k Value (The Optimal Parameter)

K refers to the number of neighbors to consider for classification. The number should be odd, to avoid a tie. However, choosing k has to be done with care. If the value of k is small, it will cause low bias and high variance; resulting in an over fitting model. Similarly, if the value of k is very large, it will lead to high bias and low variance; resulting in an under fitting model.

Most researchers suggest that the value of K should be the square root of the total number of data points, n, above 100 (Prasath *et al.*, 2017; Zhang, 2016; Cheng *et al*, 2014). This is illustrated in Equation 2.13.

$$K = \sqrt{n} \quad \text{Equation 2.13}$$

Commonly used values of K as input include 3, 5, and 9. However, Cheng *et al.* (2014) is of the opinion that adopting Equation 2.13 is not to the best interest of all, as it does not hold in all cases. The paper further argued that one of the methods of classifying test data using fixed K value gives a serious drawback, as the values of the test data might differ from one another. The paper rather suggested that the value of K should be data driven. That is, it should be decided by the distribution of data. (Cheng *et al.*, 2014).

Cross validation is another technique used to determine the value of K, however, this too has the drawback of not considering the correlation of samples. Correlation coefficient measures the strength (norm) and direction of linear relationship between two numeric variables x and y.

2.9.1.11 K-NN and Distance Measure

K-Nearest Neighbor algorithm classifies an unlabeled test sample based on the majority of similar samples among the K nearest neighbors that are closest to the test sample (Prasath *et al.*, 2017). The distance between the test sample and each of the training samples in memory is determined by a specific distance measure. Figure 16 illustrates KNN classification with K=3 and K=5 values using Euclidean distance measure.

In Figure 2.16 there are two classes: stars and triangles, the test sample is represented by a filled circle.

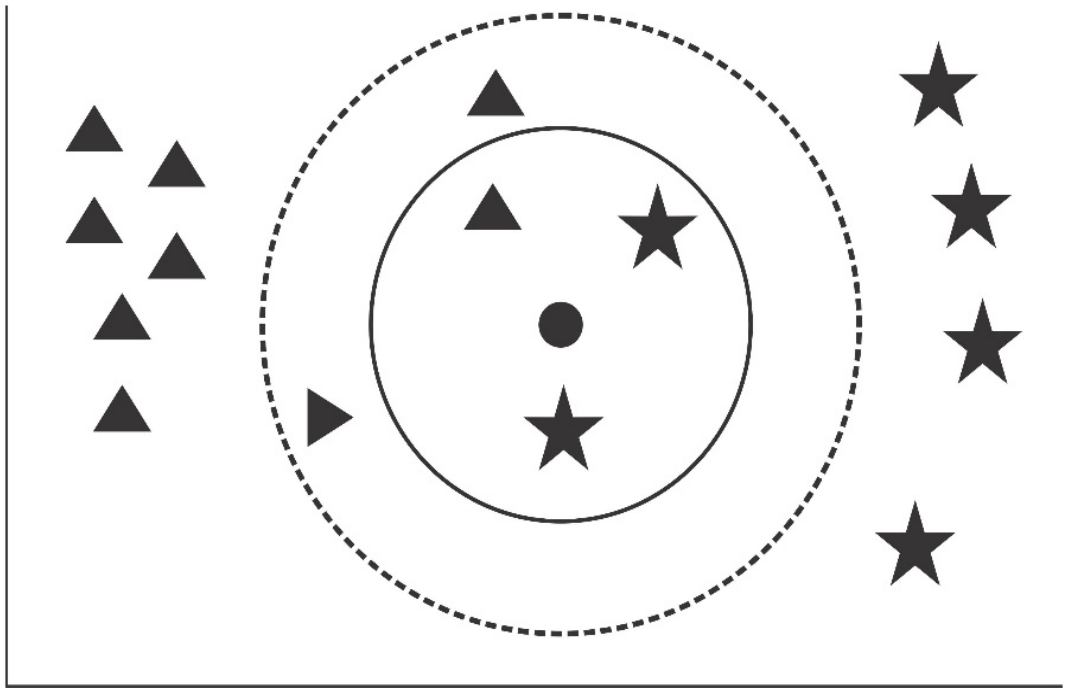


Figure 2.16: K-NN Classification with k=3 and k=5 using Euclidean Distance Measure

Source: Researcher (2024)

Finding the majority class among the K nearest neighbors predicts the class of the test sample. In Figure 2.16, where K=3, the test sample classifies to the star class, as it has the majority vote; but where K=5, the test sample classifies to the triangle class since it has the majority vote. It is noted that the dotted circle houses five objects (including both stars and triangles, while the inner circle houses three objects surrounding the filled circle test object. Algorithm to implement KNN is illustrated in subsection 2.9.1.12

2.9.1.12 K-NN Algorithm

The following steps depicts the process of measuring KNN distance to the target object:

- **i.** Load the data into the program (Application software implementing KNN)
- **ii.** Initialize the number of neighbors to be considered, K, which must be odd value.
- **iii.** To each entry or data point (or tuple) in the data file, do this:
    - a Calculate the distance between the data point (tuple) to be classified (test sample) and each data point in the training data file in memory
    - b Then add the computed distances in the training data file (i.e., add the column measures of each record)
    - c Sort the computed data points in ascending order by distances
    - d Pick the first K entries from the sorted data
    - e Observe the class of the majority vote (labels) and assign the test sample to it.

2.9.2 Naïve Bayes (NB) (Kohavi, 1996)

This soft computing algorithm relies on the Bayes theorem in its classification. It can be used for both binary and multi-class problems. Naïve Bayes method evaluates the probability of each feature independently, regardless of any correlation, and makes the

prediction based on Bayes theorem. This algorithm is based on the concept of class probabilities and conditional probabilities. A class probability is the probability of a class in the dataset. In other words, if we select a random item from the dataset, this is the probability of it belonging to a certain class. Conditional probability is the probability of the feature value, given the class.

Class probability is calculated as the number of samples in the class divided by the total number of samples, as illustrated in Equation 2.14

$$p(C) = \frac{\text{Count (instances in C)},}{\text{Count (instances in N total)}} \qquad \text{Equation 2.14}$$

Conditional probabilities are calculated as the frequency of each attribute value divided by the frequency of instances of that class. This is shown in Equation 2.15

$$p(V|C) = \frac{\text{Count (instances with V and C)}}{\text{Count (instances with V)}} \qquad \text{Equation 2.15}$$

Given the probabilities, we can calculate the probability of an instance belonging to a class, and take decision based on Bayes theorem, as shown in Equation 2.16

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \qquad \text{Equation 2.16}$$

The probabilities of the items belonging to all classes are then compared and the class with the highest probability is selected. This method is simple and easy to understand. It also performs well on noisy or irrelevant data features. Its main drawback is that each feature is treated independently, although in most cases this cannot be true (Narudin *et al.,* 2014; Nasteski, 2017; Berrar, 2018).

2.9.3 Decision Tree (J48) (Kotsiantis, 2007)

Decision trees are algorithms that have the structure of a tree. Training data set is used for the creation of the tree, which is subsequently used to make predictions on the unseen dataset (new data set). In this algorithm, the goal is to achieve the most accurate

results with the least number of decisions that must be made. The dataset used is tabulated.

The training set constitutes a table with a number of observations (rows) with various pieces of information about them called columns. One of these columns is the dependent variable, sometimes called ‘target’ or ‘label’ or ‘class’ variable. The other variables are the independent variables or features. The columns are usually many; say 80, out of which 79 will be independent variables, while only one will be the dependent variable. There is also test dataset, which also has observations with the same number of independent variables as that of the training dataset, but has no dependent variable; it is the job of the trained model to be able to predict the dependent variable in the test dataset as accurately as possible.

Therefore, the model, when fitted into the training dataset, learns the relationship between the independent variables and the dependent variable, and use that knowledge to predict the dependent (or target) variable for the test data set as accurately as possible.

Figure 2.17 illustrates a decision tree with a dataset (max-depth) of three.

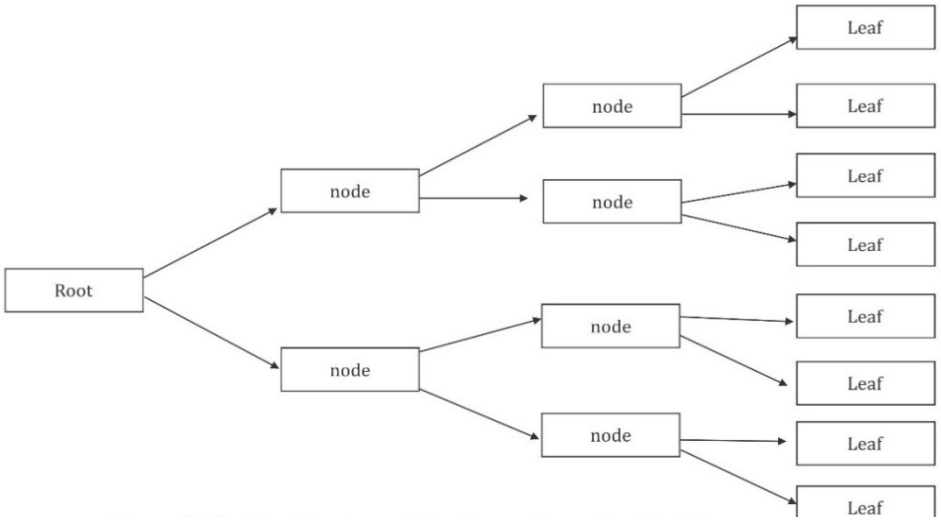


Figure 2.17: Decision Tree with a Dataset of Three Depth Three
Source: Kotsiantis (2007)

In each node, there are the following elements:

i. The node's split rule (the independent variable and its value)
ii. The Mean Square Error (MSE) of all the observations in that node
iii. Sample, being the number of observations in that node – The size of the group
iv. The target (dependent variable) – This is computed using natural logarithm

As the tree expands, it is observed that the MSE value decreases.

The common algorithm for decision trees is iterative dichotomizer 3 (ID3). It relies on the concepts of entropy and information gain. Entropy refers to the level of uncertainty in the data content.

Decision Tree has the following advantages:

i. It is simple
ii. It can deal with large dataset
iii. It can handle noise in the dataset
iv. It operates in a 'white box', unlike other algorithms like SVM and ANN. That is, it is transparent, and one can see clearly how the outcome is obtained.
v. It works with categorical variables

The disadvantages of Decision Trees include

i. It cannot predict numerical values
ii. It does not support online learning. That is, you have to rebuild your tree whenever you have new data set.
iii. Decision tree easily over fits
iv. Decision trees are slow, especially if they are complex with many branches.

2.8.4 Random Forest (RF) (Breiman, 2001)

In soft computing, there are meta-models that combine the predictions of several smaller models to generate a better model that will provide a final prediction from data it did not train with. This is called “ensemble learning”. Several decision trees are often combined together in an ensemble method called “bootstrap aggregation” or “Bagging”. The resulting aggregation is called Random Forest. Random forests are simple but effective. When being fitted to a training dataset, many decision trees are constructed, with each tree being fitted in a random subset of the data.

To generate a prediction for a new observation, the Random Forest simply averages the predictions of all its trees and returns that as its prediction. This works well because the bootstrap sampling and the feature subset are meant to make the trees as uncorrelated as possible (although they are all still based on the same dataset and feature set), allowing each tree to discover slightly different relationship in the data. This results in their average having less variance – fewer overflows – than any single tree, and therefore better generalization and prediction. It is observed that each tree is structured differently on different dataset. Hence, each tree will define similarity in a different way and predict a different value. Therefore, for a given unseen observation, the average of all the trees is basically the average of all the values of a lot of observations in the training set, which are somehow similar to it. It is a machine learning classifier frequently used in malware detection. One consequence about Random Forest is, its prediction is very poor when the test dataset is different in some fundamental way from the training set – different range of values.

2.9.5 Logistic Regression

Logistic regression is a supervised learning algorithm that solves classification problems. It is used for predicting categorical dependent variable using a given set of independent variables. The outcome of its prediction is either categorical or numeric

(discrete) values such as YES or NO, True or False, 0 or 1. However, it gives a probabilistic value that lies within the range [0, 1].

Logistic regression differs from linear regression. While logistic regression solves classification problems, linear regression solves regression problems, which are continuous in nature. Logistic regression fits an "S" shape sigmoid function, which is represented in Equation 2.17

$$\log\left[\frac{y}{1-y}\right] = b_o + b_1x_1 + b_2x_2 + b_3x_3, + \ldots + b_nx_n \qquad \text{Equation 2.17}$$

Figure 2.18 depicts the "S" shape sigmoid function.

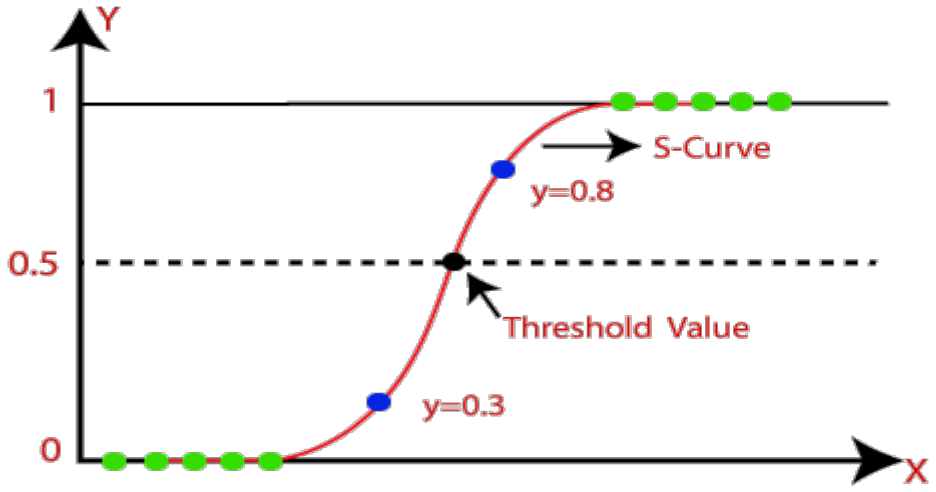


Figure 2.18: A Plot of Logistic Regression Sigmoid Function

In logistic regression, the concept of threshold value is used, and it defines the probability of either 0 or 1. Value above the threshold tends to 1, while those below the threshold tends to 0.

There are three types of logistic regression: binomial, multinomial, and ordinal logistic regressions.

i. Binomial LR has only two possible types of dependent variable, such as 0 or 1, True or False, Yes or No.

ii. Multinomial LR has a dependent variable with three or more possible unordered values such as Cat, Goad, or Sheep; train, car, tram, or byke (with transport as the dependent variable).

iii. Ordered LR has dependent variable with ordered values such as low, medium or high; red, blue, green (for color dependent variable).

Logistic regression can be applied in areas such as

i. Forecasting the effects or impact of specific changes

ii. Forecasting trends and future values

iii. Determining the strengths of different predictors

Advantages of Logistic Regression:

i. It is relatively fast compared to other supervised learning algorithms, such as SVM, K-NN, and ANN.

ii. It is easier to implement than other forms of ML algorithms

iii. It works well for cases where the dataset is linearly separable

iv. It provides useful insights. That is, it not only provides the magnitude and relevance of the independent variable, but its direction of the relationship with the dependent variable.

v. It works well with fairly large dataset

vi. It does not over fit, however, it over fits in non-linear spaces.

Disadvantages of Logistic regression:

i. It has poor accuracy

ii. It tends to underperform when the decision boundary is non-linear

iii. It does not work well with small dataset, which could result in over fitting

iv. It works well only when the dependent variable is categorical or dichotomous

v. It assumes linearity between predicted (dependent) variables and the independent variables, this is not so in real world situations.

2.9.6 Artificial Neural Networks (ANN) (Hagan et al., 1996)

Artificial Neural network is inspired by the human brain. However, it takes much more time during training (Hira and Gillies, 2015; Aljawarneh *et al.,* 2017), as such, it is difficult to apply on handheld mobile phones where real-time is a constraint. But multilayer perceptron (MLP) is commonly used because it has medium level complexity. ANN is flexible and supports high degree of complexity, but it is complex and hard to interpret.

2.10 Data Collection

The data that is used in this research work is obtained from Kaggle repository; it is Network Security Laboratory - Knowledge Discovery in Databases (NSL-KDD) dataset. It is a balanced dataset with normal data type, and four anomalous data types: denial of service (DOS) data type, probe or surveillance data type, user to root (U2R) data type, and remote to local (R2L) data type (GitHub Inc. 2020). Table 2.9 illustrates the anomalous attack types

Table 2.9: Attack Types

s/n	Attack names	Attack Types
1	Back	DOS
2	Buffer-overflow	U2R
3	ftp_write	R2L
4	Guess_passwd	R2L
5	Imap	R2L
6	Ipsweep	Probe
7	Land	DOS
8	Loadmodule	U2R

9	Multihop	R2L
10	Neptune	DOS
11	Nmap	Probe
12	Perl	U2R
13	Phf	R2L
14	Pod	DOS
15	Portsweep	Probe
16	Rootkit	U2R
17	Satan	Probe
18	Smurf	DOS
19	Spy	R2L
20	Teardrop	DOS
21	Warezclient	R2L
22	Warezmaster	R2L

(Source: GitHub Inc., 2020)

The data obtained consists of two types of records or classes: normal (Benign) and anomaly (Malicious) with columns or features contained in attribute relations file format (ARFF).

The steps used to collect data from Kaggle repository include:

i. Download and collect benign and malicious applications
ii. Decompress (unzip) the applications to extract content using APK tool
iii. Extract permission request features from each application using read manifest.exe
iv. Build the dataset in an ARFF format and load in the computer device used.
v. Normalize the dataset and split into training and test datasets in the ratio of 80: 20
vi. Train algorithms using the training dataset
vii. Test the trained model for generalization using test dataset.

viii. Carry out performance evaluation on the tested dataset using confusion matrix.

ARFF file is an ASCII text file that describes the list of instances that shares a set of attributes. It is divided into two: header subsection and data subsection. The header describes the features of the dataset, while the data section describes the observations of the dataset as illustrated in Figure 2.19.

```
%       The Header
%       Title:      NSL-KDD Data set
%       Source:     GitHub Inc. (2020)
%
%
@RELATION       NSL-KDD Data set

@ATTRIBUTE      “duration”                      real
@ATTRIBUTE      “protocol_type”                 {tcp, udp, icmp}

@ATTRIBUTE      “service”                       {aol, auth, bgp, courier
…}
@ATTRIBUTE      “flag”                          {OTH, REJ, RSTO, RSTR
...}
@ATTRIBUTE      “src_bytes”                     real
@ATTRIBUTE      “dst_bytes”                     real
@ATTRIBUTE      “land”                          {‘0’, ‘1’}
@ATTRIBUTE      “wrong_fragment”                real
@ATTRIBUTE      “urgent”                        real
@ATTRIBUTE      “hot”                           real
@ATTRIBUTE      “num_failed_logins”             real
@ATTRIBUTE      “logged_in”                     {‘0’, ‘1’}
@ATTRIBUTE      “num_compromised”               real
@ATTRIBUTE      “root_shell”                    real
@ATTRIBUTE      “su_attempted”                  real
@ATTRIBUTE      “num_root”                      real
@ATTRIBUTE      “num_file_creations”            real
@ATTRIBUTE      “num_shells”                    real
@ATTRIBUTE      “num_access_files”              real
@ATTRIBUTE      “num_outbound_cmds”             real
@ATTRIBUTE      “is_host_login”                 {‘0’, ‘1’}
@ATTRIBUTE      “is_guest_login”                {‘0’, ‘1’}
@ATTRIBUTE      “count”                         real
@ATTRIBUTE      “srv_count”                     real
@ATTRIBUTE      “serror_rate”                   real
@ATTRIBUTE      “srv_serror_rate”               real
@ATTRIBUTE      “rerror_rate”                   real
@ATTRIBUTE      “srv_rerror_rate”               real
```

@ATTRIBUTE	“same_srv_rate”	real
@ATTRIBUTE	“diff_srv_rate”	real
@ATTRIBUTE	“srv_diff_host_rate”	real
@ATTRIBUTE	“dst_host_count”	real
@ATTRIBUTE	“dst_host_srv_count”	real
@ATTRIBUTE	“dst_host_same_srv_rate”	real
@ATTRIBUTE	“dst_host_diff_srv_rate”	real
@ATTRIBUTE	“dst_host_same_src_port-rate”	real
@ATTRIBUTE	“dst_host_srv_diff_host_rate	real
@ATTRIBUTE	“dst_host_serror_rate”	real
@ATTRIBUTE	“dst_host_srv_serror_rate”	real
@ATTRIBUTE	“dst_host_rerror_rate”	real
@ATTRIBUTE	“dst_host_srv_rerror_rate”	real
@ATTRIBUTE	“class”	{‘normal’, ‘anomaly’}

% *The Body of ARFF*

@DATA

0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00,normal,20

0,udp,other,SF,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,1,0.00,0.00,0.00,0.00,0.08,0.15,0.00,255,1,0.00,0.60,0.88,0.00,0.00,0.00,0.00,0.00,normal,15

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,123,6,1.00,1.00,0.00,0.00,0.05,0.07,0.00,255,26,0.10,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19

0,tcp,http,SF,232,8153,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,5,0.20,0.20,0.00,0.00,1.00,0.00,0.00,30,255,1.00,0.00,0.03,0.04,0.03,0.01,0.00,0.01,normal,21

0,tcp,http,SF,199,420,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,30,32,0.00,0.00,0.00,0.00,1.00,0.00,0.09,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,121,19,0.00,0.00,1.00,1.00,0.16,0.06,0.00,255,19,0.07,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,166,9,1.00,1.00,0.00,0.00,0.05,0.06,0.00,255,9,0.04,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,117,16,1.00,1.00,0.00,0.00,0.14,0.06,0.00,255,15,0.06,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,remote_job,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,270,23,1.00,1.00,0.00,0.00,0.09,0.05,0.00,255,23,0.09,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,133,8,1.00,1.00,0.00,0.00,0.06,0.06,0.00,255,13,0.05,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,205,12,0.00,0.00,1.00,1.00,0.06,0.06,0.00,255,12,0.05,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,199,3,1.00,1.00,0.00,0.00,0.02,0.06,0.00,255,13,0.05,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21 *

0,tcp,http,SF,287,2251,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0.00,0.00,0.00,0.00,1.00,0.00,
0.43,8,219,1.00,0.00,0.12,0.03,0.00,0.00,0.00,0.00,normal,21

0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,2,20,1.00,0.00,1.00,0.20,0.00,0.00,0.00,0.00,warezclient,15

0,tcp,name,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,233,1,1.00,1.00,0.00,0.00,0.00,0.06,0.0
0,255,1,0.00,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19

0,tcp,netbios_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,96,16,1.00,1.00,0.00,0.00,0.17,0.0
5,0.00,255,2,0.01,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,tcp,http,SF,300,13788,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,9,0.00,0.11,0.00,0.00,1.00,0.00
,0.22,91,255,1.00,0.00,0.01,0.02,0.00,0.00,0.00,0.00,normal,21

0,icmp,eco_i,SF,18,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.
00,1,16,1.00,0.00,1.00,1.00,0.00,0.00,0.00,0.00,ipsweep,18

0,tcp,http,SF,233,616,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.00,0.
00,66,255,1.00,0.00,0.02,0.03,0.00,0.00,0.02,0.00,normal,21

0,tcp,http,SF,343,1178,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,10,0.00,0.00,0.00,0.00,1.00,0.00
,0.20,157,255,1.00,0.00,0.01,0.04,0.00,0.00,0.00,0.00,normal,21

0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,223,23,1.00,1.00,0.00,0.00,0.10,0.05,0.0
0,255,23,0.09,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,280,17,1.00,1.00,0.00,0.00,0.06,0.05,
0.00,238,17,0.07,0.06,0.00,0.00,0.99,1.00,0.00,0.00,neptune,18

0,tcp,http,SF,253,11905,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,10,0.00,0.00,0.00,0.00,1.00,0.0
0,0.20,87,255,1.00,0.00,0.01,0.02,0.00,0.00,0.00,0.00,normal,21

5607,udp,other,SF,147,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,255,1,0.00,0.85,1.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,248,2,1.00,1.00,0.00,0.00,0.01,0.06,0.00
,255,2,0.01,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19

507,tcp,telnet,SF,437,14421,0,0,0,0,0,1,3,0,0,0,0,0,1,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,255,25,0.10,0.05,0.00,0.00,0.53,0.00,0.02,0.16,normal,20

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,279,7,1.00,1.00,0.00,0.00,0.03,0.06,0.
00,255,13,0.05,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,http,SF,227,6588,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,22,0.00,0.00,0.00,0.00,1.00,0.00
,0.18,43,255,1.00,0.00,0.02,0.14,0.00,0.00,0.56,0.57,normal,21

0,tcp,http,SF,215,10499,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,14,14,0.00,0.00,0.00,0.00,1.00,0.
00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,241,1400,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,33,33,0.00,0.00,0.03,0.03,1.00,0.0
0,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,12,0.00,0.00,0.00,0.00,1.00,0.00,1.
00,1,53,1.00,0.00,1.00,0.51,0.00,0.00,0.00,0.00,ipsweep,17

0,tcp,finger,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,57,16,1.00,1.00,0.00,0.00,0.28,0.07,0.
00,255,59,0.23,0.04,0.01,0.00,1.00,1.00,0.00,0.00,neptune,20

0,tcp,http,SF,303,555,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,9,0.00,0.00,0.00,0.00,1.00,0.00,0.00,9,255,1.00,0.00,0.11,0.01,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,1,0.00,0.00,1.00,1.00,0.50,1.00,0.00,255,1,0.00,0.31,0.28,0.00,0.00,0.00,0.29,1.00,portsweep,20

0,udp,domain_u,SF,45,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,181,181,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,250,0.98,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,18

1,udp,private,SF,105,147,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,41,5,0.12,0.05,0.05,0.00,0.00,0.00,0.00,0.00,normal,19

0,udp,domain_u,SF,43,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,122,202,0.00,0.00,0.00,0.00,1.00,0.00,0.01,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,normal,18

0,tcp,supdup,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,205,17,1.00,1.00,0.00,0.00,0.08,0.06,0.00,255,17,0.07,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19

0,tcp,http,SF,324,2302,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,22,28,0.00,0.00,0.00,0.00,1.00,0.00,0.14,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,uucp_path,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,228,13,1.00,1.00,0.00,0.00,0.06,0.07,0.00,255,13,0.05,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,20

0,tcp,ftp_data,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,50,11,1.00,1.00,0.00,0.00,0.22,0.08,0.00,255,63,0.25,0.02,0.01,0.00,1.00,1.00,0.00,0.00,neptune,19

0,tcp,Z39_50,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,262,12,1.00,1.00,0.00,0.00,0.05,0.06,0.00,255,9,0.04,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

2,tcp,smtp,SF,1591,372,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.00,1.00,0.00,1.00,181,147,0.81,0.02,0.01,0.00,0.00,0.00,0.00,0.00,normal,21

9052,udp,other,SF,146,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,2,0.00,0.00,0.00,0.00,0.50,0.50,0.00,255,2,0.01,0.66,0.99,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,290,3006,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,11,11,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,csnet_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,108,11,1.00,1.00,0.00,0.00,0.10,0.08,0.00,255,11,0.04,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,80,80,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,80,0.31,0.02,0.31,0.00,0.00,0.00,0.00,0.00,teardrop,16

0,tcp,http,SF,255,861,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,44,44,0.02,0.02,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,4,38,1.00,0.00,1.00,0.18,0.00,0.00,0.00,0.00,warezclient,12

0,tcp,ftp_data,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,258,17,1.00,1.00,0.00,0.00,0.07,0.05,0.00,255,5,0.02,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,tcp,http,SF,302,498,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,10,0.00,0.00,0.00,0.00,1.00,0.00,0.00,131,255,1.00,0.00,0.01,0.04,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,61,4,0.00,0.00,1.00,1.00,0.07,0.10,0.00,255,4,0.02,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,2,0.01,0.02,0.01,0.00,0.00,0.00,0.77,0.00,teardrop,15

0,tcp,http,SF,220,1398,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,26,42,0.00,0.00,0.00,0.00,1.00,0.00,0.05,26,255,1.00,0.00,0.04,0.03,0.00,0.00,0.00,0.00,normal,21

0,udp,domain_u,SF,43,69,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,120,120,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,245,0.96,0.01,0.01,0.00,0.00,0.00,0.00,0.00,normal,18

0,udp,domain_u,SF,44,133,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,73,75,0.00,0.00,0.00,0.00,1.00,0.00,0.03,122,212,0.88,0.02,0.88,0.01,0.00,0.00,0.08,0.00,normal,21

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,15,0.00,0.00,0.00,0.00,1.00,0.00,1.00,2,46,1.00,0.00,1.00,0.26,0.00,0.00,0.00,0.00,nmap,17

0,tcp,uucp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,135,9,1.00,1.00,0.00,0.00,0.07,0.06,0.00,255,11,0.04,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,20

0,tcp,finger,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,24,12,1.00,1.00,0.00,0.00,0.50,0.08,0.00,255,59,0.23,0.04,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,udp,domain_u,SF,43,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,148,228,0.00,0.00,0.00,0.00,1.00,0.00,0.01,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,normal,18

0,tcp,ftp_data,SF,641,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,175,48,0.25,0.02,0.25,0.04,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,206,4,0.00,0.00,1.00,1.00,0.02,0.06,0.00,255,4,0.02,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,175,1,0.10,0.00,0.89,1.00,0.01,1.00,0.00,255,1,0.00,0.84,0.00,0.00,0.07,0.00,0.62,1.00,satan,18

0,tcp,http,SF,310,2794,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,4,4,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,63,10,0.00,0.00,1.00,1.00,0.16,0.08,0.00,255,10,0.04,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,smtp,SF,696,333,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,109,133,0.39,0.04,0.01,0.02,0.00,0.00,0.00,0.00,normal,21

0,tcp,netbios_dgm,RSTR,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,1.00,1.00,1.00,0.00,0.00,236,1,0.00,0.58,0.58,0.00,0.00,0.00,0.58,1.00,portsweep,14

0,tcp,http,SF,209,442,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,26,26,0,0,0,0,1,0,0,235,255,1,0,0,0.02,0,0,0,0,normal,21

5,tcp,pop_3,SF,26,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,233,214,0.92,0.01,0,0,0,0,0.04,0,guess_passwd *

0,tcp,http,SF,327,294,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,10,0,0,0,0,1,0,0.2,255,255,1,0,0,0,0,0,0,normal,21

4,tcp,pop_3,SF,32,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,161,0.63,0.02,0,0,0,0,0.13,0,guess_passwd,15 *

0,udp,private,SF,52,54,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0,0,0,0,1,0,0,25
5,255,1,0,0.87,0,0,0,0,0,normal,17 *

0,tcp,http,SF,295,527,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0,0,0,0,1,0,0,124,255,
1,0,0.01,0.03,0,0,0,0,normal,21

0,tcp,http,SF,316,356,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,4,4,0,0,0,0,1,0,0,38,255,1,
0,0.03,0.02,0,0,0,0,normal,

0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,10,10,0,0,0,0,1,0,0,35,10,
0.29,0.11,0.29,0,0,0,0,0,teardrop,11

3,tcp,pop_3,SF,30,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,247,0
.97,0.01,0,0,0,0,0.02,0,guess_passwd,18 *

1,tcp,telnet,RSTO,123,178,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,1,1,0,0,255,
12,0.05,0.01,0,0,0,0,0.03,0.67,guess_passwd,13 *

0,tcp,http,SF,311,294,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,27,29,0,0,0,0,1,0,0.1,255,
255,1,0,0,0,0,0,0,0,normal,21 *

0,tcp,http,SF,227,406,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,34,0,0,0,0,1,0,0.09,64,2
55,1,0,0.02,0.03,0,0,0,0,normal,21

0,udp,private,SF,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,5,5,0,0,0,0,1,0,0,255,253,
0.99,0.01,0,0,0,0,0,0,snmpguess,13 *

0,tcp,http,SF,314,358,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,10,0,0,0,0,1,0,0,231,25
5,1,0,0,0.02,0,0,0,0,normal,21

0,tcp,http,SF,243,269,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,21,21,0,0,0,0,1,0,0,255,25
5,1,0,0,0,0,0,0,0,normal,21

0,tcp,ftp_data,SF,19,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,122,15,0.
04,0.05,0.04,0.13,0,0,0,0,normal,21 *

210,tcp,telnet,SF,126,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,2
48,0.97,0.01,0,0,0,0,0.02,0.02,guess_passwd,16

0,tcp,http,SF,203,454,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,11,0,0,0,0,1,0,0.18,255
,255,1,0,0,0,0,0,0,0,normal,21

0,tcp,finger,SF,5,381,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,2,0.0
1,0.03,0,0,0.02,0,0.7,0,normal,16 *

0,tcp,http,SF,279,526,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,13,0,0,0,0,1,0,0.15,2,25
5,1,0,0.5,0.02,0,0,0,0,normal,21

0,udp,private,SF,42,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,255,253,
0.99,0.01,0.01,0,0,0,0,0,snmpguess,14 *

1,tcp,telnet,SF,24,715,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0.67,0.33,0.33,1,0.
67,255,67,0.26,0.02,0,0,0,0.01,0.7,0.69,mscan,11 *

0,udp,domain_u,SF,46,86,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,248,246,0,0,0,0,0.99,0
.01,0,255,254,1,0.01,0,0,0,0,0,0,normal,18 *

4,tcp,pop_3,SF,28,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,90,77,0.8
4,0.03,0.01,0.03,0,0,0.11,0,guess_passwd,7

4,tcp,pop_3,SF,25,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,70,67,0.9
4,0.03,0.01,0.03,0,0,0,0,guess_passwd,6

0,udp,private,SF,49,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,4,0,0,0,0,1,0,0,255,254,
1,0.01,0.01,0,0,0,0,0,snmpguess,16

2,tcp,imap4,RSTO,0,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,6,0,0,0.67,1,0.33,1,1,5
1,54,0.27,0.1,0.02,0.04,0,0,0.92,0.26,mscan,12 *

0,tcp,telnet,SF,125,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,109
,0.43,0.02,0,0,0,0,0.01,0.03,guess_passwd,10

4,tcp,pop_3,SF,31,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,208,0
.82,0.02,0,0,0,0,0.07,0,guess_passwd,18

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,27,0,0,0,0,1,0,1,2,60,1,0,1
,0.5,0,0,0,0,ipsweep,17 *

0,udp,domain_u,SF,46,77,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,188,188,0,0,0,0,1,0,0,
255,255,1,0,0.01,0,0,0,0,0,normal,18 *

0,udp,domain_u,SF,45,82,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,159,160,0,0,0,0,0.99,0
.01,0.02,255,250,0.98,0.01,0,0,0,0,0,0,normal,18 *

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,21,0,0,0,0,1,0,1,2,42,1,0,1
,0.26,0,0,0,0,nmap,16

0,tcp,ftp_data,SF,43,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,255,62,0.
24,0.02,0.01,0,0.87,0.97,0,0,normal,14 *

0,udp,private,SF,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,156,146,0,0,0,0,0.94,0.01,0
,255,208,0.82,0.03,1,0,0,0,0,0,satan

0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,5,17,1,0
,1,0.12,0,0,0,0,warezclient,13 *

0,tcp,finger,SF,9,196,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,5,5,0,0,0,0,1,0,0,155,7,0.0
5,0.06,0.01,0,0.01,0,0,0,satan,0 *

0,icmp,ecr_i,SF,520,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,184,184,0,0,0,0,1,0,0,255,
255,1,0,1,0,0,0,0,0,smurf,18 *

0,icmp,eco_i,SF,20,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,3,0.0
1,0.02,0.01,0,0,0,0,0,satan,6 *

1,tcp,ftp,RSTO,0,133,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,1,1,0,0,2,2,0.5,1,0.5,1,0,0,1,1,ipsweep,9 *

0,udp,domain_u,SF,34,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,0,0,0,0,1,0,1,213,22,0.1,0.03,0.1,0,0,0,0,0,normal,21 *

0,icmp,ecr_i,SF,1480,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,1,25,1,0,1,0.52,0,0,0,0,pod,17

0,tcp,pop_3,RSTO,0,36,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0,0,1,1,0.33,1,1,189,60,0.24,0.03,0.01,0.03,0,0,0.88,0.98,mscan,15 *

0,tcp,ftp,SF,26,157,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,1,0,0,155,71,0.46,0.03,0.01,0,0,0,0,0,guess_passwd,7

0,tcp,telnet,S3,0,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,1,0,0,255,117,0.46,0.02,0,0,0.53,0.37,0.01,0.01,processtable,18 *

0,udp,private,SF,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,254,1,0.01,0,0,0,0,0,0,snmpguess,12 *

0,tcp,telnet,RSTO,124,188,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,1,1,0,0,255,254,1,0.01,0,0,0.01,0.01,0.02,0.02,guess_passwd,10

8169,tcp,telnet,SF,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,19,0.07,0.82,0,0,0,0,0.8,0,processtable,12 *

0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,192,191,0.99,0.01,0.01,0,0,0,0,0,snmpgetattack,2 *

1,tcp,telnet,RSTO,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,7,0,0,1,0.57,1,0,0.57,96,87,0.34,0.05,0.01,0.02,0,0,0.9,0.7,mscan,11

0,tcp,pop_3,SF,30,217,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,2,0.01,0.02,0,0,0,0,0,0,guess_passwd,4

0,udp,other,SF,23,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,1,0,0.03,0,0,0,0,0,0,multihop,9

0,tcp,telnet,SF,123,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,255,49,0.19,0.02,0,0,0,0,0.01,0.04,guess_passwd,11

0,udp,private,SF,105,147,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,237,235,0.99,0.01,0,0,0,0,0,0,snmpgetattack,1

,tcp,telnet,SF,6,54,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,5,0,0.2,0.5,0.4,0.5,0.75,0.6,255,83,0.33,0.1,0,0,0,0,0.33,0.48,mscan,11

0,udp,private,SF,47,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,255,
1,0,0.01,0,0,0,0,0,snmpguess,15

0,tcp,telnet,SF,122,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,25,
0.1,0.02,0,0,0,0,0,0.04,guess_passwd,9

0,tcp,telnet,SF,120,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,115
,0.45,0.02,0,0,0,0,0.01,0.03,guess_passwd,11

0,icmp,ecr_i,SF,508,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,182,182,0,0,0,0,1,0,0,255,
255,1,0,1,0,0,0,0,0,smurf,18 *

1,tcp,smtp,SF,2599,293,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0,0,1,0,0,255,13
8,0.54,0.15,0,0,0,0,0.44,0,mailbomb,11

0,udp,other,SF,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,140,1,0,0,0,0,0.01,0.01,0,8,2,
0.25,0.62,0.25,0,0,0,0.12,0,satan,10

0,icmp,eco_i,SF,20,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,55,0,0,0,0,1,0,1,3,47,1,0,
1,0.26,0,0,0,0,saint,15

0,udp,private,SF,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0,0,1,0,0,255,255,
1,0,0.02,0,0,0,0,0,snmpguess,16 *

8,tcp,ftp,SF,220,688,0,0,0,4,0,1,0,0,0,0,4,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,53,27,0.51,0.08,0.02,0.00,0.00,0.00,0.00,0.00,buffer_overflow,1
*

0,udp,private,SF,47,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,15

0,tcp,ftp_data,SF,12,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,7,25,1.00,0.00,1.00,0.12,0.00,0.00,0.00,0.00,warezmaster,12

0,tcp,ftp,SF,26,157,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,255,129,0.51,0.03,0.00,0.00,0.00,0.00,0.00,0.00,guess_passwd,9

0,udp,private,SF,46,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,6,6,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,snmpguess,15
*

282,tcp,ftp,SF,163,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1
.00,0.00,0.00,255,5,0.02,0.24,0.00,0.00,0.34,0.00,0.24,0.00,warezmaster,15

0,udp,other,SF,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,145,1,0.00,0.00,0.00,0.00,0.0
1,1.00,0.00,255,1,0.00,0.62,0.99,0.00,0.00,0.00,0.00,0.00,satan,19

282,tcp,ftp,SF,161,597,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1
.00,0.00,0.00,255,97,0.38,0.03,0.00,0.00,0.00,0.00,0.00,0.00,warezmaster,13

282,tcp,ftp,SF,164,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1
.00,0.00,0.00,255,31,0.12,0.03,0.00,0.00,0.40,0.00,0.02,0.00,warezmaster,15

,tcp,ftp,SF,162,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,17,0.07,0.02,0.00,0.00,0.14,0.00,0.25,0.00,warezmaster,12

282,tcp,ftp,SF,162,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,57,0.22,0.03,0.00,0.00,0.13,0.00,0.02,0.00,warezmaster,11

1179,udp,other,SF,146,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,33,14,0.42,0.12,1.00,0.00,0.00,0.00,0.00,0.00,snmpgetattack,0

0,udp,private,SF,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,snmpgetattack,1 *

0,udp,private,SF,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,5,5,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,15

0,icmp,ecr_i,SF,1008,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,303,303,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,smurf,18 *

282,tcp,ftp,SF,158,597,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,72,0.28,0.03,0.00,0.00,0.05,0.00,0.02,0.00,warezmaster,12

0,udp,private,SF,49,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,253,0.99,0.01,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,14

0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,253,0.99,0.01,0.01,0.00,0.00,0.00,0.00,0.00,snmpgetattack,3

4,tcp,pop_3,SF,28,103,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,214,0.84,0.02,0.00,0.00,0.00,0.00,0.04,0.01,guess_passwd,15 *

282,tcp,ftp,SF,160,597,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,50,0.20,0.02,0.00,0.00,0.00,0.00,0.55,0.00,warezmaster,14

,udp,private,SF,41,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,12 *

0,icmp,eco_i,SF,30,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,247,247,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,normal,20

0,icmp,eco_i,SF,30,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,227,227,1,0,1,0,0,0,0,normal,20 *

0,udp,domain_u,SF,35,91,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,4,0,0,0,0,1,0,0,75,255,249,0.98,0.01,0.01,0,0,0,0,0,normal,21 *

Figure 2.19: Attribute-Relation File Format (ARFF)
Source: GitHub Inc. (2020)

NSL-KDD dataset may not be a perfect dataset, but it is an improvement on the KDD'CUP 99 dataset, for it does not include redundant, noisy and missing records (Anthony, 2014; GitHub Inc, 2020). The dataset contains 262,178 attack records and 812,814 normal records in the training dataset; and 29,378 attack records and 49,911 normal records in the test dataset. These records contain forty-one (41) permission features. They are reduced to a fixed range of [0, 1] using normalization.

2.10.1 Data Preprocessing

Figure 2.20 depicts the processing techniques used in this research work. It consists of the dataset used to train and test the models, NSL-KDD dataset. The dataset is is then uploaded to the cloud via mobile host agent. In the cloud, the data is cleaned, preprocessed and extracted and balanced using SMOTE, Pearson correlation, and information gain (IG); then normalized (singh *et al.,* 2015; Ikram and Cherukuri, 2016), and selected to an appropriate size using PCA. The extracted data features are then split into training (80%) and test datasets (20%). The training dataset is used by random forest (RF), the training algorithm, which samples the data patterns and develop models to learn the patterns. The models are built from base algorithms such as decision tree (DT), K-Nearest Neighbors (K-NN), Naïve Bayes (NB) and Logistic Regression (LR). The trained models each predicts the labels in the test dataset. The predictions are then aggregated using voting classifier to a concensus model called soft vote. This aggregated soft model is trained and used to predict the labels in the test dataset into normal and malicious threats.

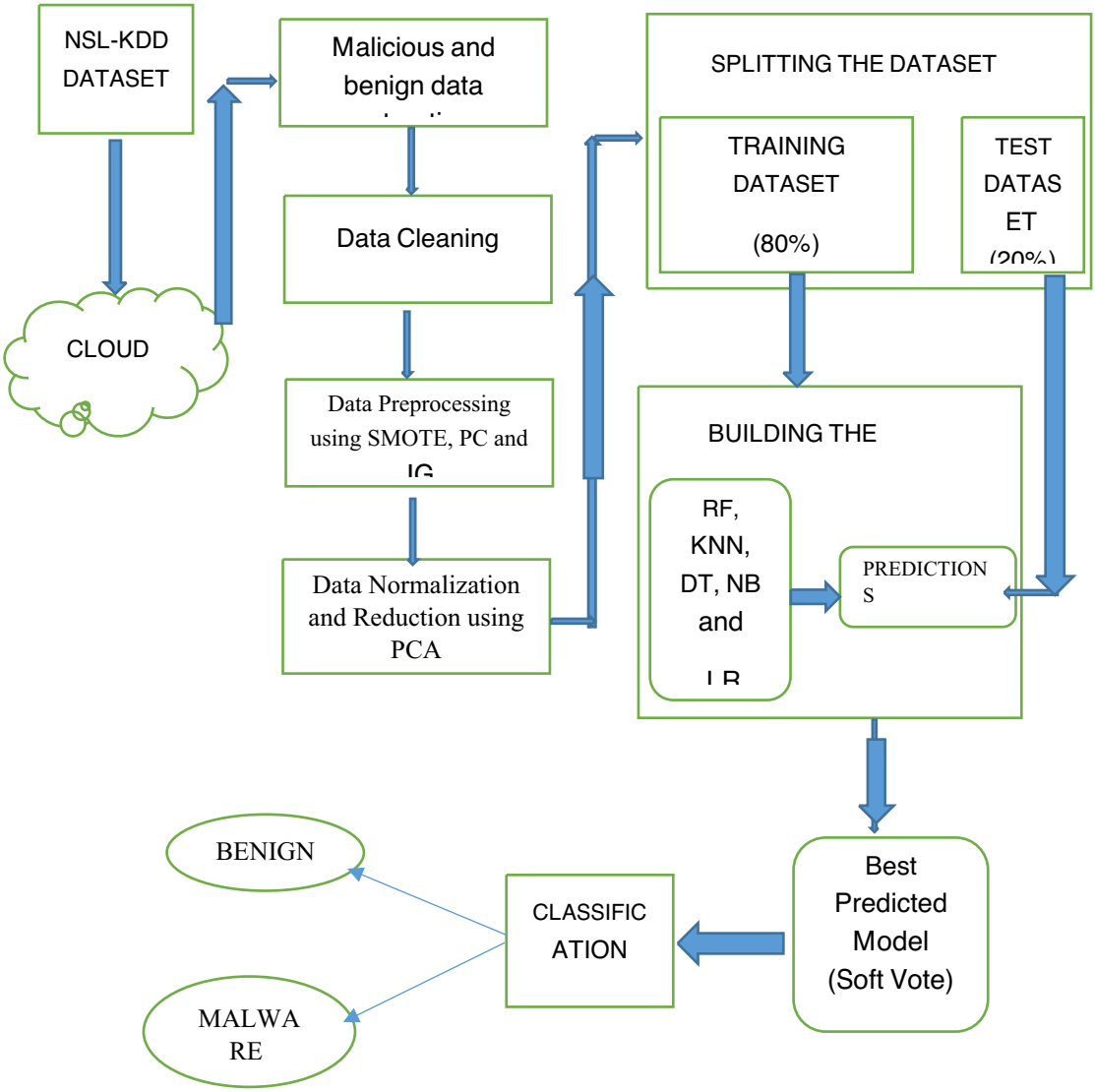


Figure 2.20: Normalization, Feature Reduction and Feature Classification
Source: Researcher (2024)

2.10.2 Min_Max Normalization

Min-Max normalization performs linear transformation on original data. Let (X_1, X_2) be min and max boundaries of an attribute (feature), and (Y_1, Y_2) be the new scale at

which normalization is done, then for V_i value of the attribute, the normalized value U_i is given in Equation 2.17.

$$U_i = \frac{Vi - X1}{X2 - X1} (Y_2 - Y_1) + Y_1 \quad \text{Equation 2.17}$$

To apply this model on the NSL-KDD dataset, features from the ARFF file are used, which not only define the attributes but also present the observations in comma separated values (CSV) format. The significant features to use without distorting information flow are described in Table 2.10. These significant features are extracted during the computation of principal components (CPs) by PCA using Eigen values and Eigenvectors.

Table 2.10: Significant Features

S/N	Attribute Name	Description
1	Src_bytes	This is the number of data bytes that is transferred from source to destination in a single connection.
2	Count	This is the number of connections to the same destination host, as the current connection in the past two seconds
3	Srv_count	This is the number of connections to the same service (port number), as the current connection in the past two seconds.
4	Logged_in	This indicates the log in status: 1 if successfully logged in, and 0 otherwise
5	Num_compromised	This is the number of compromised conditions, especially network systems infected and turned into zombies, then added into the botnet family. Another example is the “SQL injection” where malicious code is embedded into a normal application to corrupt it.
6	Dst_host_count	This is the number of connections having the same destination host IP address

7	Num_outbound_cmd	This is the number of outbound commands issued by command and control (C&C) servers in a file transfer (ftp) protocol session to steal data, customer contact list, and carry out nefarious acts like SEND_SMS, READ_CONTACT, WRITE_CALL_LOG.

These parameters are applied in Equation 2.17 where

X_1: is the *src_byte* feature, and is the initial number of bytes sent to the destination host

X_2: is the *count* feature, which is the number of times the destination host is connected

Y_1: is the initial *logged_in* feature, which is usually zero (0)

Y_2: is the successfully *logged in* status, which is one (1)

V_i: is the *destination host count* with the same IP address

U_i: is the number of *compromised* systems in a connection session of not more than two Seconds.

Using Neptune, a DOS attack type, with the following extracted features from ARFFfile

$X_1 = 0$;

$X_2 = 199$

$Y_1 = 0$

$Y_2 = 1$,

$V_i = 255$,

U_i: is the number of compromised systems being computed.

Therefore,

$$Ui = \frac{255 - 0}{199 - 0}(1 - 0) + 0 = 1.30$$

That is, the number of systems compromised and added to the botnet family, within the two seconds DOS attack window is one.

Using ipsweep, a probe attack type, the number of compromised systems include:

$X_1 = 18$;

$X_2 = 1$,

$Y_1 = 0$,

$Y_2 = 1$,

$V_i = 1,$

$$Ui = \frac{1 - 18}{1 - 18}(1 - 0) + 0 = \frac{-17}{-17} = +1$$

Since the result is one, then the compromised systems are within the range

To compute the number of local systems compromised by a remote attacker (R2L), warezclient attack type was used, with the following features from ARFF file.

$X_1 = 334;$

$X_2 = 2,$

$Y_1 = 0,$

$Y_2 = 1,$

$V_i = 4$

$$Ui = \frac{4 - 334}{2 - 334}(1 - 0) + 0 = \frac{-330}{-332} = 0.99$$

The compromised system in this case is 0.99, which is within the range.

To rank and properly generalize the model, the entire dataset needs to be cleaned, filtered and the dimensions reduced

2.10.3 Dimensionality Reduction

Dimensionality refers to the number of features in a dataset. When the number of features in a dataset is very large relative to the number of observations, some algorithms struggle to train effective models. This is called *curse of dimensionality.*

Current trends have it that high volumes of data are generated, and can contain noise and redundant data. These irrelevant data need to be removed for effective learning of the patterns. The process is called dimensionality reduction. It is a process of reducing the number of random variables under consideration, and obtaining a set of relevant variables. Dimensionality reduction is divided into feature selection and feature extraction (Hira and Gillies, 2015):

i. **Feature Selection**: In this type of dimensionality reduction, a subset of the original features is found that can be used to model the problem at

hand. There are three ways to achieve the subset: by filtering, by wrapping, and by embedding.

- **A filter:** A Filter selects features based on their relationship with the target. Filters can be active or passive and constitute four main types: low-pass, high-pass, notch and band-reject.
- **Wrapper methods** are based on greedy search algorithm as they evaluate all combinations of the features and select the combination that produces the best result for the specific ML algorithm.
- **In Embedded technique,** malicious code like virus is infiltrated into a legitimate code; and executes each time the legitimate app is executed (Jiang *et al.*, 2020).

  Feature selection filters irrelevant, redundant and noisy data from the original data set. Some algorithms used for the purpose include decision trees (DT), random forests, genetic algorithms and principal component analysis (PCA).

ii **Feature Extraction –**Feature extraction creates new smaller set of features that still captures most of the useful information. While feature selection keeps a subset of the original dataset, feature extraction creates entirely new ones. One such example is deep learning. Feature extraction can be supervised such as linear dimensional analysis (LDA) or unsupervised (PCA).

2.10.3.1 Principal Component Analysis (PCA)

Principal component analysis is a dimension reduction tool, which reduces a large set of variables to a small set without distorting information flow in the large dataset

(Wallisch, 2014). It is a mathematical procedure that transforms a number of correlated variables into a smaller number of uncorrelated (orthogonal) variables called principal components. It increases interpretability, but at the same time minimize information loss and successively maximize variance (Wallisch, 2014).

The first principal component accounts for as much of the variability in the dataset as possible, and each succeeding principal component accounts for as much of the remaining variability as possible, and so on. Variability is achieved using Eigen vectors and Eigen values. An Eigen vector is a unit vector pointing in the direction of the new coordinate axis, and the axis with the highest Eigen value explains the most variation (Wallisch, 2014). Traditionally, PCA is performed on a square symmetric matrix, which can be covariance matrix or singular value decomposition (SVD) or correlation matrix. Correlation matrix is used if the variance of the individual variant differs much or if the individual variants differ.

The Goals of PCA.

The goals of CPA include:

i. Finding the relationships between observations
ii. Extracting the most important information from data
iii. To detect outliers and remove them
iv. To reduce the dimension of the data by keeping only relevant information.
v. It ensures that original variables have the highest correlation (relationship) with the principal components.

These goals are achieved by finding the PCA space, which represents the maximum variance of a given data (Jolliffe and Cadima, 2016; Holland, 2019). To appreciate the works of PCA, certain terms used in its extraction process are defined.

Definition of PCA Relevant Terms:

The Mean

The mean of a data set helps us to ascertain the spread of a given data.

Let X = [1, 2, 4, 6, 12, 15, 25, 45, 65, 67, 98]

The mean of X is $\tilde{x}$ and it is represented in Equation 2.18.

$$\bar{x} = \sum_{i=1}^{n} \frac{xi}{n}, \forall i = 1, 2, 3, \ldots, n \qquad \text{Equation 2.18}$$

In the above sample X,

$$\sum_{i=1}^{n} X = 1 + 2 + 4 + 6 + 12 + 15 + 25 + 45 + 65 + 67 + 98 = 340$$

$$\bar{x} = \frac{340}{11} = 30.9 = 31.$$

Standard Deviation (σ)

The standard deviation, (σ), of a dataset also measures the spread of the dataset. It is the average distance of the mean of a dataset to a point. It is represented or computed using Equation 2.19.

$$\sigma = \sqrt{\sum_{i=1}^{n} \frac{(xi - \bar{x})2}{(n - 1)}} \qquad \text{Equation 2.19}$$

Division by n-1 is due to the fact that samples are used rather than the entire population.

As an illustration, let two samples X and Y be:

X = [0, 8, 12, 20] and Y = [8, 9, 11, 12] as illustrated in Table 2.11

Table 2.11: Parameters to Compute Standard Deviation (σ)

X	$X - \bar{x}$	$(X - \bar{x})^2$
0	-10	100
8	-2	4
12	2	4
20	10	100
Total: **40**		**208**

Mean $(\bar{x}) = \frac{40}{4} = 10$

$$\sigma = \sqrt{\left(\frac{208}{3}\right)} = \sqrt{69.33} = 8.33$$

The computation of $\bar{y}$ and **σ** in Y = [8, 9, 11, 12] is illustrated in Table 2.12

Table 2.12: Illustrating the Spread of Data

Y	$Y - \bar{y}$	$(Y - \bar{y})^2$
8	-2	4
9	-1	1
11	1	1
12	2	4
Total: 40		10

Mean $(\bar{y}) = \frac{40}{4} = 10$

$$\sigma = \sqrt{\sum_{i=1}^{n} \frac{(yi - \bar{y})2}{(n-1)}} = \sqrt{(\frac{10}{3})} = \sqrt{3.33} = 1.82$$

Observe that the spread of the set X from the mean is wider than that of the set Y as its σ_x is 8.3, whereas that of Y, σ_y, is 1.82. That is, the difference between the mean and the value of X or Y is wider in X than that of Y.

Variance (σ^2)

The variance is another measure of the spread of data from the mean. The formula is illustrated in Equation 2.20

$$\text{Var (x)} = \sigma^2 = \sum_{i=1}^{n} \frac{(xi - \bar{x})2}{(n-1)} \qquad \text{Equation 2.20}$$

Covariance (cov (x, y))

Covariance is always measured between two dimensional spaces, unlike standard deviation and variance which operate on single dimensional space. If you have a dataset of 3-dimension in R^3, such as (x, y, z), you can measure the covariance of x and y, or x and z or y and z. The covariance of a dimension and itself results in the variance of that dimension. That is, cov (x, x), cov (y, y), and cov (z, z) result in the variance of x, y, and z respectively. Equation 2.21 can be rewritten as

$$\sigma^2 = \sum_{i=1}^{n} \frac{(xi - \bar{x})(xi - \bar{x})}{(n-1)} \qquad \textbf{Equation 2.21}$$

The formula for covariance is similar to that of variance, except that it deals with two dimensions at a time, rather than one dimension used by variance and standard deviation. It is illustrated in Equation 2.23.

$$\textbf{Cov (x, y) =} \sum_{i=1}^{n} \frac{(xi - \bar{x})(yi - \bar{y})}{(n-1)} \qquad \text{Equation 2.23}$$

What is more important in Equation 2.23 is whether or not the value of covariance is positive (+ve) or negative (-ve). If the value is positive, then the two dimensions increase together; but if it is negative, then as one-dimension increases, the other is decreasing. However, if the covariance value is zero, then the two dimensions are independent of each other. Since covariance is measured between any two dimensions, for n-dimensional space, covariance values are computed using Equation 2.24.

$$C^{nxn} = \frac{n!}{(n-2)! * 2} \qquad \text{Equation 2.24}$$

A useful way to compute all the possible covariance values between all dimensions is to use matrices. The definition of covariance matrix is illustrated in Equation 2.25.

$$C^{n \times n} = (C_{ij}, C_{ij}) = \text{cov}(\text{Dim}_i, \text{Dim}_j) \qquad \text{Equation 2.25}$$

Where:

$C^{n \times n}$: is a matrix with n rows and n columns;

$\text{Dim}_{i,j}$: is the ith and jth dimensions

For example, 3-dimension in R^3, such as (x, y, z) is illustrated in Figure 2.21

$$C_{x,y,z} = \begin{pmatrix} \text{Cov}(x,x) & \text{cov}(x,y) & \text{cov}(x,z) \\ \text{Cov}(y,x) & \text{cov}(y,y) & \text{cov}(y,z) \\ \text{Cov}(z,x) & \text{cov}(z,y) & \text{cov}(z,z) \end{pmatrix}$$

Figure 2.21: Covariance Matrix

The entry in row 2, column 3 in Figure 2.22 is the covariance value computed between the 2nd and 3rd dimensions. We have already seen that the covariance of a dimension with itself produces the variance of that dimension. - cov (x, x); cov (y, y) and cov (z, z) are variances of x, y, and z respectively. Since covariance matrix is commutative, ie, cov (a, b) = cov (b, a), then the matrix is symmetric about the main diagonal. That is, the diagonal values of the covariance matrix represent the variance of the variables – x, y and z.

Matrices

A matrix is a rectangular array of numbers. The numbers there in are called entries. Most of the time, matrices are bordered by square brackets, [ ]. The size of the matrix is described in terms of rows and columns. For example, in a 3 x 3 matrix, such as

$$A = \begin{pmatrix} e & x & -\sqrt{2} \\ 0 & ½ & 1 \\ 0 & 0 & 0 \end{pmatrix},$$

there are three rows and three columns**.** Other types of matrices include:

Row vector, for example, a 1 x 4 vector is [2 1 0 -1], which has one row and four columns.

A column vector, say a 2 x 1 matrix has two rows and one column, $\begin{bmatrix} 1 \\ 3 \end{bmatrix}$

A unit vector, has both one row and one column, 1 x 1 matrix, [4]

A 3 x 2 matrix has three rows and two columns, $\begin{pmatrix} 1 & 2 \\ 3 & 0 \\ 1 & 4 \end{pmatrix}$

(Anton and Rorres, 2014).

Capital letters are used to denote matrices, while lower case letters denote numerical quantities. For example,

$$A = \begin{pmatrix} 2 & 1 & 7 \\ 3 & 4 & 2 \end{pmatrix}$$

$$\mathbf{C} = \begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix}$$

In matrix, numerical quantities are called scalars. Let **A** be the matrix; and in **A**, a_{11}, a_{12} … a_{34} are numerical quantities.

$$A = \begin{pmatrix} a11 & a12 & a13 & a14 \\ a21 & a22 & a23 & a24 \\ a31 & a32 & a33 & a34 \end{pmatrix}, \text{ A is a 3 x 4 matrix}$$

Transpose of a matrix

The transpose of a matrix **A,** denoted by $\mathbf{A^T}$, is the matrix A with rows and columns interchanged. That is, the first column of transpose $\mathbf{A^T}$ is the first row of **A;** the second column of transpose $\mathbf{A^T}$ is the second row of **A,** and so on. Transpose $\mathbf{A^T}$ is represented below.

$$A^T = \begin{pmatrix} a11 & a21 & a31 \\ a12 & a22 & a32 \\ a13 & a23 & a33 \\ a14 & a24 & a34 \end{pmatrix}, \ A^T \text{ is a 4 x 3 matrix}$$

Identity matrix

A square matrix with 1s on the main diagonal and zeros elsewhere is called an identity matrix.

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The identity matrix is denoted by **I**. If **A** is any m x n matrix, then the product of **A** and the identity matrix, **I,** is shown in Equation 2.26.

$$\mathbf{AI = A} \text{ and } \mathbf{IA = A} \qquad \text{Equation 2.26}$$

Inverse of a matrix

If **A** is a square matrix, and if a matrix **B** of the same size can be found such that

$$\mathbf{AB = BA = I,}$$

then **A** is invertible (or nonsingular) and **B** is called an inverse of **A.** If **B** cannot be found, then **A** is said to be singular.

An invertible matrix

An invertible matrix (non-zero matrix) has exactly one inverse, and the product of **A** and its inverse, $\mathbf{A^{-1}}$ results in an identity matrix, **I,** as shown in Equation 2.27.

$$\mathbf{AA^{-1} = A^{-1}A = I} \qquad \text{Equation 2.27}$$

Matrix A is invertible iff the product of the entries on the main diagonal minus the product of the entries off the main diagonal is not equal to zero. That is, ad - bc $\neq 0$.

The inverse of a matrix **A** is given by Equation 2.28

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\mathbf{A^{-1}} = \begin{bmatrix} a & -b \\ -c & d \end{bmatrix} \qquad \text{Equation 2.28}$$

To multiply two matrices together, the number of rows of one matrix must be equal to the number of columns of the other matrix. That is, if **A** is an m x r matrix and **B** is an r x n matrix, then **AB =BA = m x n.** The product entry is determined by multiplying each element of a row in matrix **A** by a corresponding element in the column of matrix **B** and add them up. For example, multiply a 2 x 3 matrix by a 3 x 4 matrix.

$$A = \begin{pmatrix} 1 & 2 & 4 \\ 2 & 6 & 0 \end{pmatrix}$$

and

$$B = \begin{pmatrix} 4 & 1 & 4 & 3 \\ 0 & -1 & 3 & 1 \\ 2 & 7 & 5 & 2 \end{pmatrix}$$

$$AB = \begin{pmatrix} 1 & 2 & 4 \\ 2 & 6 & 0 \end{pmatrix} \begin{pmatrix} 4 & 1 & 4 & 3 \\ 0 & -1 & 3 & 1 \\ 2 & 7 & 5 & 2 \end{pmatrix}$$

Matrix multiplication requires that the number of columns of **A** be equal to the number of rows of **B;** however, multiplying a square matrix with a vector results in another vector. For example, a square matrix **D** and a column vector **V**, when multiplied produces another vector.

$$\mathbf{D} = \begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix}$$

$$\mathbf{V} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

$$\mathbf{DV} = \begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 12 \\ 8 \end{bmatrix} = 4 \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

As can be seen, the resulting vector is 4 times the original vector. Therefore, the vector

$\begin{bmatrix} 3 \\ 2 \end{bmatrix}$ is an Eigen vector, and the number 4 is the Eigen value

Determinant

The determinant of a 2 x 2 matrix **A**, is the product of the entries of the main diagonal minus the product of the entries off the main diagonal, as illustrated in Equation 2.29.

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\mathbf{det\ (A) = ad - bc} \qquad \text{Equation 2.29}$$

Eigen vectors and Eigen values

Eigen vectors and Eigen values are used to determine the principal components in PCA process and filter out the less important variables in training dataset. Let **A** be n x n matrix and let $\mathbf{x} \in \mathbb{R}^n$ (where n represents the level of dimensions) be a non-zero vector,

if the product of the matrix and the vector is equal to the product of a scalar, λ**,** and the vector as illustrated in Equation 2.30,

$$\mathbf{Ax} = \lambda\mathbf{x} \qquad \text{Equation 2.30}$$

Then λ is called an Eigen value of the matrix **A**; and **x** is called an Eigen vector of **A** associated with λ**.** The set of all Eigen values of an n x n matrix **A** is denoted by $\sigma(A)$ referred to as the spectrum of **A**. When an Eigen vector **X** is multiplied by **A**, the resulting vector is in the same direction as **X** or opposite to it. From Equation 2.31.

$$\mathbf{Ax} = \lambda\mathbf{x}$$

$$\mathbf{Ax} - \lambda\mathbf{x} = \mathbf{0} \qquad \text{Equation 2.31}$$

To ensure that we are dealing with matrices, we multiply λ**X** by **I.** As illustrated in Equation 2.32. That is,

$$\mathbf{Ax} = \lambda\mathbf{Ix}$$

$$\mathbf{Ax} - \lambda\mathbf{Ix} = \mathbf{0} \qquad \text{Equation 2.32}$$

$$(\mathbf{A} - \lambda\mathbf{I})\,\mathbf{x} = \mathbf{0}$$

Or

$$(\lambda\mathbf{I} - \mathbf{A})\,\mathbf{x} = \mathbf{0}$$

The matrix used to compute Eigen values and Eigen vectors should not be invertible but a singular matrix. Therefore, the determinant of a singular matrix is equal to zero as shown in Equation 2.33.

$$\text{Det}\,((\lambda\mathbf{I} - \mathbf{A}) = \mathbf{0} \qquad \text{Equation 2.33}$$

The expression $((\lambda\mathbf{I} - \mathbf{A})\,\mathbf{X} = \mathbf{0})$ is a polynomial called the characteristic polynomial of **A, w**hile $\mathbf{det}\,((\lambda\mathbf{I} - \mathbf{A}) = \mathbf{0})$ is called the characteristic equation**.** It therefore means that the roots of the characteristic polynomial are the Eigen values of **A** (Anton and Rorres, 2014). While Eigen vectors should not be zero, Eigen values can be zero. For example,

$$\text{Let A} = \begin{pmatrix} 0 & 5 & -10 \\ 0 & 22 & 16 \\ 0 & -9 & -2 \end{pmatrix} \quad \text{and} \quad X = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\text{Therefore AX} = \begin{pmatrix} 0 & 5 & -10 \\ 0 & 22 & 16 \\ 0 & -9 & -2 \end{pmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = 0 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

(Anton and Rorres, 2014**).**

Given the importance of Eigen values and Eigen vectors to PCA in the determination of Principal components, it becomes necessary to illustrate how they are computed.

Let $A = \begin{pmatrix} -2 & -4 & 2 \\ -2 & 1 & 2 \\ 4 & 2 & 5 \end{pmatrix}$ and let **I** be a 3 x 3 identity matrix

$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, then $\lambda \mathbf{I} = \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$

Now

Det $\mathbf{(A - \lambda I) = 0}$

$$\text{Det}\left[\begin{pmatrix} -2 & -4 & 2 \\ -2 & 1 & 2 \\ 4 & 2 & 5 \end{pmatrix} - \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}\right] = \begin{pmatrix} -2-\lambda & -4 & 2 \\ -2 & 1-\lambda & 2 \\ 4 & 2 & 5-\lambda \end{pmatrix} = 0$$

There are many ways to solve or expand this determinant. It could be done using *reduced row echelon* form, *or row factor* form. In this example, we will use the row factor form. The resulting quadratic equations are illustrated in Equation 2.33, Equation 2.34, and Equation 2.35

$$\text{Det}\begin{pmatrix} -2-\lambda & -4 & 2 \\ -2 & 1-\lambda & 2 \\ 4 & 2 & 5-\lambda \end{pmatrix} = (-2-\lambda) \begin{bmatrix} 1-\lambda & 2 \\ 2 & 5-\lambda \end{bmatrix}$$

$$= (-2-\lambda)\,[(1-\lambda)(5-\lambda) - 2 \text{ x } 2]$$

$$= (-2-\lambda)\,(5 - \lambda - 5\lambda + \lambda^2 - 4)$$

$$= (-2-\lambda)\,(-6\lambda + \lambda^2 + 1)$$

$$= (-2-\lambda)\,(\lambda^2 - 6\lambda + 1)$$

$$= -2\lambda^2 + 12\lambda - 2 - \lambda^3 + 6\lambda^2 - \lambda$$

$$= -\lambda^3 + 11\lambda + 4\lambda^2 - 2$$

$$= -\lambda^3 + 4\lambda^2 + 11\lambda - 2 \qquad \text{Equation 2.34}$$

$$\text{Det}\begin{pmatrix} -2-\lambda & -4 & 2 \\ -2 & 1-\lambda & 2 \\ 4 & 2 & 5-\lambda \end{pmatrix} = -(-4)\begin{bmatrix} -2 & 2 \\ 4 & 5-\lambda \end{bmatrix}$$

$$= 4[(-2)(5-\lambda) - 2x4]$$

$$= 4[(-2)(5-\lambda) - 8]$$

$$= 4(-10 + 2\lambda - 8)$$

$$= 4(2\lambda - 18)$$

$$= 8\lambda - 72 \qquad \text{Equation 2.35}$$

$$\text{Det}\begin{pmatrix} -2-\lambda & -4 & 2 \\ -2 & 1-\lambda & 2 \\ 4 & 2 & 5-\lambda \end{pmatrix} = 2\begin{bmatrix} -2 & 1-\lambda \\ 4 & 2 \end{bmatrix}$$

$$= 2[(-2) \times 2 - 4(1-\lambda)]$$

$$= 2(-4 - 4 + 4\lambda)$$

$$= 2(-8 + 4\lambda)$$

$$= -16 + 8\lambda$$

$$= 8\lambda - 16 \qquad \text{Equation 2.36}$$

Summing up Equation 2.36, Equation 2.37 and Equation 238, we have Equation 2.39.

$$[-\lambda^3 + 4\lambda^2 + 11\lambda - 2] + [8\lambda - 72] + [8\lambda - 16] = 0$$

$$-\lambda^3 + 4\lambda^2 + 11\lambda - 2 + 8\lambda - 72 + 8\lambda - 16 = 0$$

$$-\lambda^3 + 4\lambda^2 + 27\lambda - 90 = 0$$

$$\lambda^3 - 4\lambda^2 - 27\lambda + 90 = 0 \qquad \text{Equation 2.37}$$

Equation 2.38 is a quadratic equation in the third degree. To solve it, we first need to find the factors of 90, which are:

$\pm 1, \pm 2, \pm 3, \pm 5, \pm 6, \pm 9, \pm 10, \pm 15, \pm 18, \pm 30, \pm 45$, and ± 90

We need to find a factor of 90 that when plugged into Equation 2.92 will result in zero (0). By trial and error, $3^3 - 4 \times 3^2 - 27 \times 3 + 90 = 0$

$= 27 – 36 – 81 + 90 = 0$

Therefore $(\lambda$ **- 3)** is a factor of the quadratic equation. Hence

$$\lambda^3 - 4\lambda^2 - 27\lambda + 90 = 0$$

$$(\lambda\text{-}3)\ (\lambda^2 - \lambda - 30) \qquad \text{Equation 2.38}$$

From the second-degree quadratic equation, $(\lambda^2 - \lambda - 30)$, the factors are numbers that when multiplied together gives you -30, and when added together gives you the coefficient of λ. That is, 5 x (-6) = -30

And 5 + (-6) = -1. Therefore, from observation, the factors are $(\lambda +5)\ (\lambda - 6) = 0$

From Equation 2.84, we have

$$(\lambda - 3)\ (\lambda +5)\ (\lambda - 6) = 0$$

$(\lambda - 3) = 0$; hence $\lambda = 3$

$(\lambda +5) = 0$; hence $\lambda = -5$

$(\lambda - 6) = 0$; hence $\lambda = 6$.

Therefore, the spectrum of matrix A, $\sigma(A)$, (the set of Eigen values of the matrix A) is

$\sigma\ (A) = 3\ -5 + 6. = 4$

Now, for each Eigen value, there is a corresponding (associated) Eigen vector. We now need to find the Eigen vectors. Let the Eigen vector V be:

$$V = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

For $\lambda = 3$, and from the characteristic polynomial $(A - \lambda I)V = 0$, we have

$$\begin{pmatrix} -2-\lambda & -4 & 2 \\ -2 & 1-\lambda & 2 \\ 4 & 2 & 5-\lambda \end{pmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Substituting 3 for λ and subtract we have

$$\begin{pmatrix} -5 & -4 & 2 \\ -2 & -2 & 2 \\ 4 & 2 & 2 \end{pmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Let X = 1, then the first two equations are Equation 2.39 and Equation 2.40

$$-5 - 4Y + 2Z = 0 \qquad \text{Equation 2.39}$$

$$-2 - 2Y + 2Z = 0 \qquad \text{Equation 2.40}$$

Subtracting Equation 2.40 from Equation 2.41, we have Equation 2.42

$$3 + 2Y = 0$$

$$Y = \frac{-3}{2} \qquad \text{Equation 2.41}$$

Substitute the value of Y in Equation 2.42 and solve for Z as illustrated in Equation 2.43

$$-2 - 2Y + 2Z = 0$$

$$-2 - 2\left(\frac{-3}{2}\right) + 2Z = 0$$

$$-2 + 3 + 2Z = 0$$

$$1 + 2Z = 0$$

$$2Z = -1$$

$$Z = -\tfrac{1}{2} \qquad \text{Equation 2.42}$$

Check for correctness by substituting the values of X, Y, and Z in Equation 2.42

Therefore, the Eigen vector, V, for $\lambda = 3$ is

$$V_3 = \begin{bmatrix} 1 \\ \frac{-3}{2} \\ -\frac{1}{2} \end{bmatrix}, \text{ and scaling up V by 2 produces } \begin{bmatrix} 2 \\ -3 \\ -1 \end{bmatrix}$$

Following the same technique, the Eigen vectors for $\lambda = -5$ and $\lambda = 6$ are

$$V_{-5} = \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix} \text{ and } V_6 = \begin{bmatrix} 1 \\ 6 \\ 16 \end{bmatrix} \text{ respectively.}$$

The Eigen vector with the highest Eigen value produces the first principal component; and the Eigen vector with the second highest Eigen value produces the second principal component. Therefore, from our example,

Principal Component 1 (PC 1) has $\lambda = 6$ and

$$V_6 = \begin{bmatrix} 1 \\ 6 \\ 16 \end{bmatrix}$$

Principal Component II (PC II) has $\lambda = 3$ and

$$V_3 = \begin{bmatrix} 2 \\ -3 \\ -1 \end{bmatrix}$$

Principal Component III (PC III) has $\lambda = -5$ and it is rejected.

$$V_{-5} = \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix}$$

From the foregoing, V_6 and V_3 are selected, while V_{-5} is rejected. Normally, in situations where there are many vectors, a threshold is set, say >= 70%; and Eigen values that form 70% or more of the spectrum determine the selection of principal components of the affected vectors. Therefore, the selection is in line with the threshold.

The spectrum of **A** is (3 – 5 + 6) = 4;

$\frac{3}{4} * 100 = 75\%$, and

$\frac{6}{4} * 100 = 150\%$

$\frac{-5}{4} * 100 = -125\%$

Algorithm to Compute Eigenvectors and Eigen Values

i. From the characteristic polynomial, form the characteristic equation

$$(A - \lambda I) X = 0$$

$$\text{Det}(A - \lambda I) = 0$$

OR

$$\text{Det}(\lambda I - A) = 0$$, this form is used in most literature

Expand the determinant using row factor form (or reduced row echelon form)

ii. Solve the resulting quadratic equation to get the Eigen values of the given singular matrix A (The matrix should not be an invertible matrix)

iii. Find the corresponding Eigen vectors to the Eigen values.

iv. Sort Eigen vectors according to their corresponding Eigen values.

v. Select the Eigen vectors that have the highest Eigen values

vi. The selected Eigen vectors represent the projection of PCA.

(Jolliffe and Cadima, 2016; Holland, 2019).

PCA process is illustrated in the algorithm that follows.

Algorithm to Compute PCA Using Covariance Matrix Method

i. Get some data, say two or three dimensions

ii. Compute the mean

iii. Subtract the mean from the data points of each dimension

iv. Calculate the covariance between two or more dimensions

v. From the covariance matrix, find the Eigen vectors and Eigen values

vi. Sort Eigen vectors according to their corresponding Eigen values.

vii. Select the Eigen vectors that have the highest Eigen values

viii. The selected Eigen vectors represent the projection of PCA.

(Jolliffe and Cadima, 2016; Holland, 2019).

Eigen vectors are usually unit matrices, they are extensible; and produce the same number of Eigen values, which are multiples of the Eigen vectors.

The pre-processed data is then used by Random Forest algorithms to train base models such as SVM, KNN, DT, NB that will develop models to analyze, detect.

2.11 Training the Algorithm

In training, input is mapped to the output, over a sequence of pairs $\{(x_1,y_1), (x_2,y_2), \ldots, (x_t, y_t)\}$ and each connection has a weight (w_t) associated with it. Each x_t is an example of feature vector; $yt \in \{-1, +1\}$ is its label and wt is the weight vector. The weight describes the likelihood that the learning patterns will reflect the actual relationship in the data. At each step t during training, the algorithm makes a label prediction $h_t(x_t)$, which for linear classification is $h_t(x) = \text{sign}(w_t.x)$. After making the prediction, the

algorithm then compares the value with the label yt, if $h_t(x_t) \neq y_{t,}$ an error is recorded for time t. The algorithm maintains different confidence metrics for each feature so that less confident weights (misclassified weights) are updated more aggressively than more confident weights.

However, if there is no difference between the predicted label and the actual label $h_t(x_t)$, then learning for this data feature is complete, and there is no need to continue the training; otherwise the training algorithm updates the connection weights with the aim of reducing their differences until the expected label is achieved (Dewa and Maglaras, 2016; Nyuyen *et al.,* 2017; Dong *et al.,* 2020; Shi *et al.,* 2020; AWS, 2021).

2.11.1 Some Normal Characteristics Learned by Base Classifiers

In dynamic analysis, the base models are trained to learn the features of normal applications. This knowledge is used to assess applications on the net or host systems, and any deviation from the normal characteristics attracts the application being flanked as abnormal (malware). Some of these characteristics are inherent in NSL-KDD dataset. For instance:

i. Categorical features in ARFF file:

   (a) Protocol types used by normal applications are transmission control protocol (tcp) and user datagram protocol (udp); while malware tend to use more of internet control message protocol (icmp). Icmp not only report errors in data transmission, and diagnosis of host system performance through traceroute and ping; but it is also used to execute distributed denial of service (DDOS) attack type in the form of icmp

flood, smurf attack type with spoofed (faked) IP address, and ping_of_death attack type. Unlike tcp and udp, icmp does not require that devices must connect via three-way handshake before messages are sent; nor does it require specific port to send its messages.

(b) Services used by normal applications include http, private, smtp, ftp_data, other, telnet, and domain_u; while malware tend to use the following services private, mtp, finger, netbios_path, z39_50, and cs_net_ns. If any of these services is used in conjunction with tcp or udp and SF flag, then a malware is detected.

(c) Flag used by normal applications is safe_flag (SF); while malware tend to use S0 and REJ flags (very common with Neptune – DOS attack).

ii. POLP – Principle of least privileges (Lord, 2020). It states that "persons, applications, or processes should be given the bare minimum privileges (resources) to complete a given task". Normal applications obey POLP principle, while abnormal applications tend to request for more than necessary resources for their tasks.

iii. Android install_time permissions**:**

(a) Normal applications declare permissions from Android_manifest.xml file

(b) Malware declare permissions that include hardware such as GPS, GPRS, Camera, and external and undocumented APIs (Application Programming Interfaces, usually from third party stores)

iv Central process unit (CPU): Normal applications use CPU to process data, but malware tend to use graphics processing unit (GPU) to process data and evade detection.

v. Dangerous permissions: These permission types, which are 24 in number out of 135 approved permissions, are used by malware. They tend to violate user privacy (Raymond *et al.,* 2020; Sun *et al.,* 2016).

vi. Over privileged permissions: These permissions are found in manifest.xml file but they are not used at run_time. These types of permissions are requested for from the innocent user by malware in defiance to POLP principle for their nefarious intents.

vii. Business Transactions – Malware applications tend to eavesdrop and intercept (hijack) business transactions online; while normal applications do not.

viii. Third Party Applications - These applications are developed by other companies that clone applications (70% of these applications are malware) (Atkinson, 2015). While Google patches the vulnerabilities in normal applications regularly, third parties are reluctant to effect the patches in their versions, hence they continue to send older versions with vulnerabilities.to Google Play Store.

2.12 Ensemble Learning Methods

Ensemble learning combines several base classifiers to form one optimized predictive classifier (Pooja and Pushpalatha, 2019; Mary and Kumar, 2020; Brownlee, 2021). It can be used to decrease variance (bagging), decrease bias (boosting), and improve prediction (stacking). It is also divided into two groups:

i. Sequential learning, with different models trained sequentially (independently), and the mistakes of previous models are learned by their successors. Thus, giving the mislabeled classifiers higher weights. A good example is boosting technique.

ii. Parallel learning, where base models are trained in parallel, as in bagging technique, and their results aggregated into a stronger classifier using voting scheme (majority voting or plurality voting or hard voting).

2.12.1 Bagging (Bootstrap and Aggregation)

Bagging, also called bootstrap aggregation, seeks a diverse group of ensemble members by varying the training dataset. It is classified into bootstrapping and aggregation. Bootstrapping is a sampling technique where samples are derived from the whole population with replacement. Replacement means that if a row (record) is selected, it is returned to the training dataset for potential reselection. That is, a record may be selected zero, one, or more times for a given training dataset (Brownlee, 2021; Singh, 2018). Bootstrapping involves using a single machine learning algorithm, typically an unpruned decision tree, and training each model with a different sample of the same training dataset. Bagging reduces variance and over fitting. Example bagging algorithms include

i. Bagging decision tree
ii. Random Forest
iii. Extra trees

The predictions made by ensemble members are combined or aggregated using voting classification techniques.

Algorithm of Bagging Process

The following are the steps used to perform bagging process

i. Multiple subsets are created from the original dataset, selecting the observations (records) with replacement.

ii. A base learner (weak model) is trained on each of the data subsets

iii. The models run in parallel and are independent of each other.

iv. The final prediction is determined by combining the predictions from all the trained models using a voting process.

Figure 2.22 depicts the bagging process.

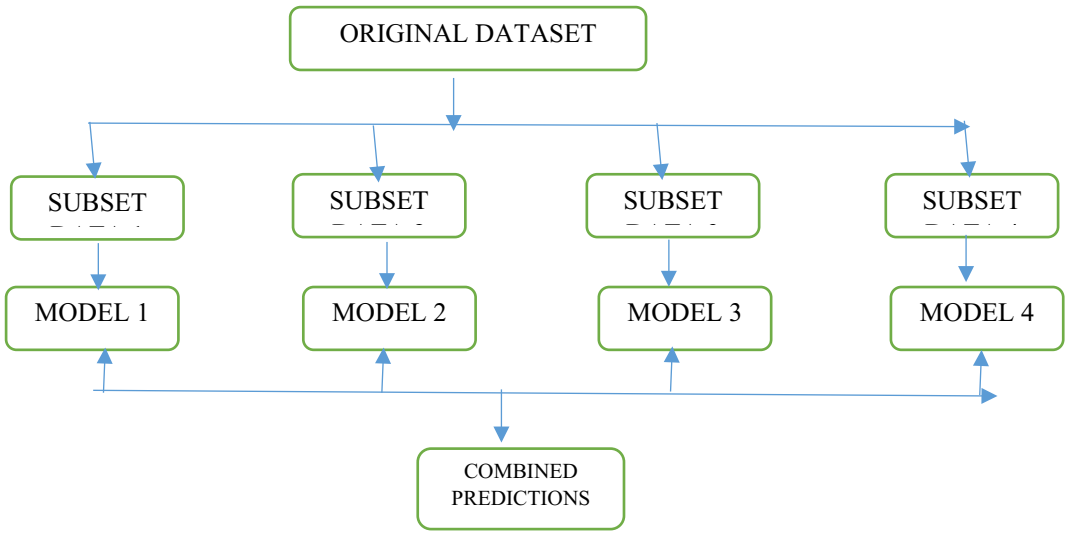


Figure 2.22: Architecture of Bagging Process
Source: Singh (2018)

2.12.2 Boosting

Boosting algorithm tries to build a strong learning (predictive) model from the mistakes of several base models. A base model is one that performs better than random guessing, but still performs poorly at assigning classes to objects. Boosting can be either binary with two labeled classes {-1, +1} (malicious or benign) respectively or multi-class.

It starts by assigning weights to the training dataset in memory, chooses and trains appropriate classifiers sequentially, focusing on misclassified results by previous models. It updates misclassified labels of previous base model. Then a second model is

trained using the updated dataset, thus reducing the errors made by the previous one. Boosting reduces bias error, which arises when models are unable to identify relevant trends in the data. Types of boosting include:

i. AdaBoost (Adaptive Boosting)
ii. Gradient Tree Boosting
iii. XGBoosting (Extreme Gradient Boosting)

Using Adaboost as boosting algorithm, with KNN, SVM, DT, and LR as base models a more formidable model is formulated by harnessing the predictions of the base models through a voting process (hard vote). The hard voting method uses the predicted labels of the classifiers and a majority rule system.

Adaboost (Freund and Schapire, 1996)

Adaboost, also called Adaptive Boosting, is a boosting method that improves the predictive capacity of base classifiers. It can combine rough or moderately accurate results to produce a more accurate result. It can function as a binary classifier, where only two classes are considered within a range {-1, 1}; and as a multi-class method, where more than two possible classes are considered. Boosting tends to solve two main problems:

i. How to adjust the training set to suit the base classifier (feature selection)
ii. How to combine the base classifiers into a strong classifier.(voting classification)

Algorithm to carry out a boosting process

The following steps are followed to carry out a boosting process

i. A subset is randomly selected from the original set
ii. Initially, all data points are given equal weights
iii. A base model is trained with this subset

iv. This model is used to make predictions on the entire dataset.

v. Errors are calculated using the actual values and predicted values.

vi. Observations which are incorrectly predicted are given higher weights.

vii. Another model is trained using the updated dataset, and predictions are made on the dataset.

viii. Similarly, multiple models are created (trained), each correcting the errors of the previous model before it.

ix. The final model (strong learner) is the weighted mean of all the weak (base) learners.

Therefore, boosting algorithm combines the weak learners to form the strong learner. Figure 2.23 depicts the architecture of boosting algorithm.

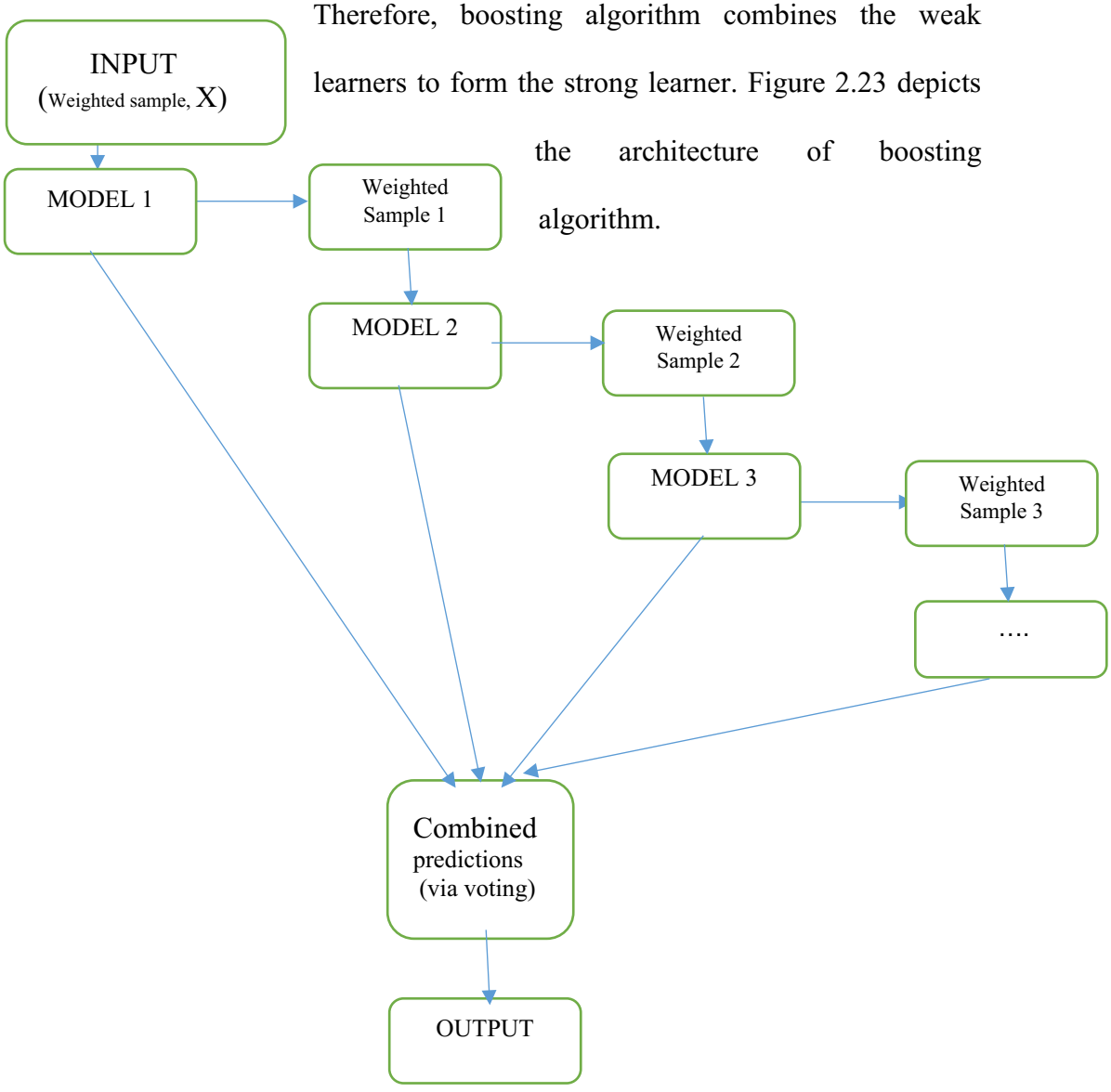


Figure 2.23: Architecture of Boosting Algorithm

Singh, 2018.

2.12.3 Stacking Ensemble Learning (Stacked Generalization)

Stacking seeks for a diverse group of base algorithms by varying the model types to fit the training dataset, and using another model to combine the predictions. In stacking, a learner is trained to combine individual learners. The individual learners are called zero-level learners; while the combiner is called first-level learner or meta-learner. In stacking, ensemble members are referred to as level-zero models, while the combiner is referred to as level-1 model. Examples of stacking process include stacked model, blending and super ensemble. Figure 2.24 depicts the architecture of Stacking process.

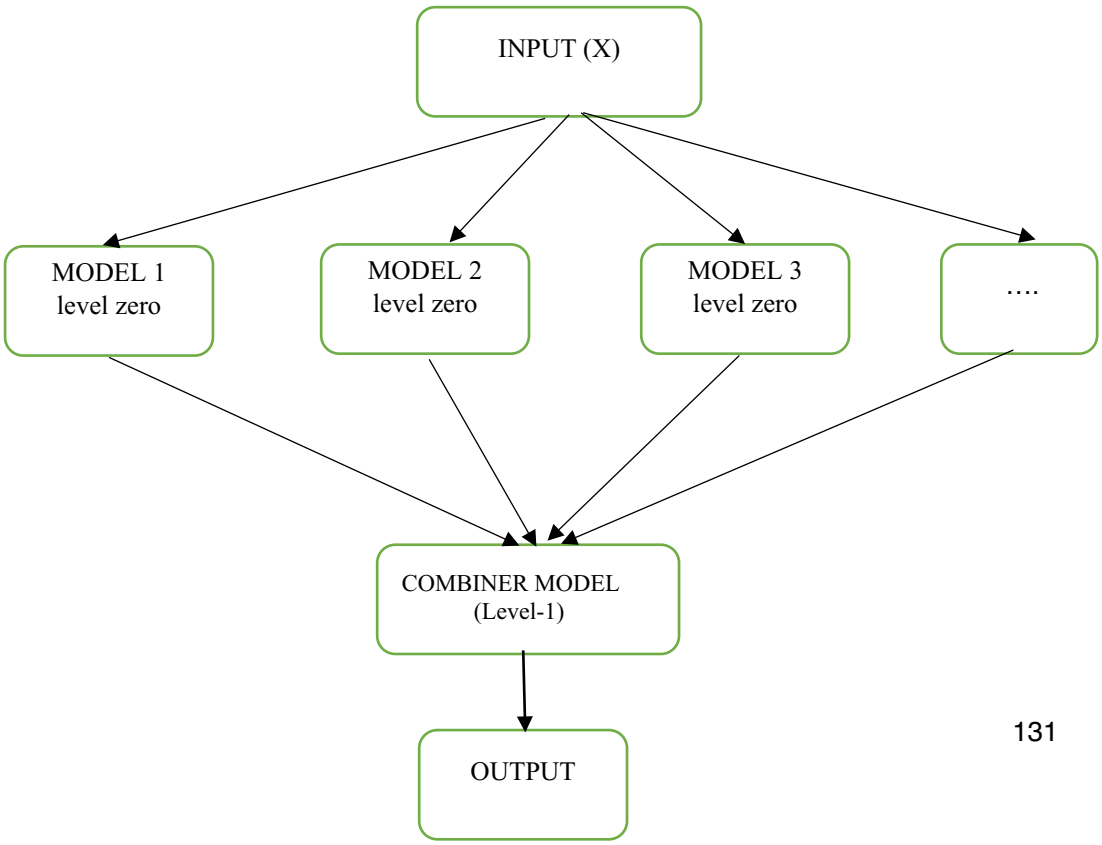


Figure 2.24: Architecture of Stacking Process.
Source: Singh, 2018

2.12.4 Voting Techniques in Ensemble Learning

In ensemble learning, different machine learning methods, working independently of each other, are moderated by the training algorithm such that they predict the class the target object is to be assigned. The training algorithm then uses a voting technique to determine the final prediction. Voting involves each model that made a prediction to assign a vote to the class that was predicted. The votes are then tallied and an outcome is chosen using the tallies. In classification, there are four types of voting techniques (Brownlee, 2021):

i. Plurality voting technique
ii. Majority voting technique
iii. Unanimous voting technique
iv. Weighted voting technique

Plurality voting selects the class label with most votes. When a tie occurs, the votes are sorted and the first one is taken for the expected prediction, instead of taking a random selection. Majority voting (Hard vote) selects the class that has more than half the votes, where no such class exists, then no prediction is made. That is, in hard voting, every individual classifier votes for a class, and the majority wins. Majority voting is best for

independent models' outputs (that is, the results of the independent predictions are diverse).

Weighted voting weighs the predictions made by each model, e.g. based on the average performance of the model, such as classification accuracy. Assigning weights to classifiers can involve using an optimization algorithm and a holdout dataset, a linear model or any machine learning model. Performance evaluation metrics can also be used to assign weight such as accuracy, precision, recall and f-measure. Weighting is done because some classifiers are more accurate than others, as such get a larger share of the vote (Brownlee, 2021).

In soft voting, every individual classifier provides a probability value that a specific data point belongs to a particular target class.

2.13 Performance Evaluation

2.13.1 Cross Validation in Soft Computing Techniques.

Cross validation is a technique used to assess how the results of statistical analysis generalize to a test dataset. That is, it provides an insight on how the trained or learned model adapts to an unseen dataset. A round of cross validation comprises the partitioning of data into complementary subsets, and then analysis is performed on all but one subset. After this, the analysis is validated on the one subset (testing set). To reduce variability, many rounds of cross validation are performed using many different partitions and then an average of the results is taken. Types of cross validation include (Berrar, 2018):

Holdout Method of Cross Validation

In this method, a part of the training data is removed and used to get predictions from the model trained on the rest of the data. The error estimation (bias) tells how the model is doing with the unseen data or the validation set. However, this method suffers from

high variance, because it is not certain which data might end up in the validation set and the results might be entirely different for different sets.

K-fold Crosses Validation

There is never enough data to train a model, therefore removing a part of it for validation poses a problem of under fitting. By reducing the training data, we risk losing important pattern/trends in the dataset, which in turn increases error induced by bias. Therefore, in K-fold cross validation, the data is divided into k subsets. The holdout method is then repeated k times, so that each time, one of the k subsets is used as the test/validation data set and the other K-1 subsets are used as the training set. The error estimation is averaged over all k trials to get the total effectiveness of the model trained. This gives a bias-variance tradeoff. That is, the process reduces bias as we are using most of the data for training, and also reduces variance as most of the data is also being used in the validation set (testing set). Generally, K can be equal to 5 or 10; but it can take any value including K=1- called leave out 1 cross validation. The data used to train and test models is stored in the desktop, laptop or mobile device from where it is ported to the cloud using mobile host agent for analysis and detection.

2.13.2 Standard Evaluation Metrics

The task of algorithms is to search for patterns in training dataset and construct mathematical models to learn those patterns. These models are then evaluated on the basis of their predictive capacities. The evaluation is either

i. 10% split cross-validation
ii. 33% split cross-validation
iii. K-fold cross validation

With 10% split cross-validation, 10% of data is used for testing, while the remaining 90% is used for training the model. For 33% split cross-validation, 33% of the data is

used for testing purpose, and the remaining 67% data set is used for training the model. The K-fold cross validation applies the classifier on the data k times, and each time with an 80:20 ratio. That is, 80% data for training and 20% data for testing. The final model is the average of the k-iterations.

Whereas, split cross-validation takes shorter time comparatively to evaluate results, it has over-fitting drawback, especially the 33% cross-validation evaluation. This drawback occurs when the classifier memorizes the training set features instead of getting trained. Because of over-fitting problem, k-fold validation produces much better results. In order to evaluate the performance of the classifiers, the following standard metrics are computed:

i. **True Positive Rate (TPR)**, which is the proportion of the correctly classified instances.

ii. **False Positive Rate (FPR)**, which is the proportion of incorrectly classified instances.

iii. **Precision**: This is the number of true positives divided by the total number of elements labeled as belonging to the positive class.

iv. **Area under the curve (AUC),** which provides the relationship between false negative and false positive is illustrated in Equation 2.43

$$\text{AUC} = \tfrac{1}{2}\left(\frac{TP}{TP+TN}+\frac{TN}{TN+FP}\right) \qquad \text{Equation 2.43}$$

AUC, which is a graphical representation of receiver operating curve (ROC), has some unanticipated problems in its application to IDS evaluation, which include: problems in detecting appropriate units for analysis; bias towards unrealistic detection approaches; and questionable presentation of false alarm data (Dewa and Maglaras, 2016; Azad and Jha 2014; Anthony, 2014). Confusion metrics characteristics used in measuring the

precision of IDS are computed in Equation 2.44, Equation 2.45, Equation 2.46, and Equation 2.47:

$$\text{TPR (Recall)} = \frac{TP}{TP + FN} \quad \text{Equation 2.44}$$

$$\text{FPR} = \frac{FP}{TN + FP} \quad \text{Equation 2.45}$$

$$\text{Precision} = \frac{TN}{TN + FP} \quad \text{Equation 2.46}$$

$$\text{F-Measure} = \frac{(2 \times \text{Recall} \times \text{Precision})}{(\text{Recall} + \text{Precision})} \quad \text{Equation.2.47}$$

Where:

i. True Positive (TP) is the number of normal data correctly classified
ii. True Negative (TN) is the number of malware samples correctly classified
iii. False Positive (FP) is the number of normal samples classified as malware
iv. False Negative (FN) is the number of malware samples classified as normal

Precision, also called **specificity**, returns the rate of relevant results, rather than irrelevant results. Recall is sensitivity for the most relevant results. F-Measure is the value that estimates the entire system performance by combining precision and Recall into a single number. The maximum number of 1.00 indicates the best result (Narudin *et al.*, 2014).

Accuracy: - This is a metric that measures how correctly an IDS works, in terms of precision of detection, number of false alarm and stability; because intrusion data set is relatively lower than normal dataset, making them harder to detect than normal dataset, it results in excessive false alarm. Accuracy measures include:

i. **Sensitivity and Specificity**: These two measures (metrics) attempt to measure the accuracy of a two-class problem. When IDS classifies data, its decision is either right or wrong.

ii **Total Delay**: This is the difference between t_{attack} and $t_{response}$. The smaller the value of total delay, the better the IDS is, with respect to its response.

Where t_{attack} is the time of attack; and $t_{response}$ is the time it takes to respond after an attack.

iii **Quality of Data**: - The quality of data is influenced by several factors:

- **Source of the data**– Data should be from reliable and appropriate sources.
- **Selection of sample** – Sample data should be unbiased
- **Sample size** – The sample size should be neither over nor under size
- **Time of data** – Data should be frequently updated in real time
- **Complexity of data** – Data should be simple.

Matthew Correlation coefficient (MCC)

MCC is another form of performance evaluation technique, it is illustrated in Equation 2.48.

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)\,(TP + FN)\,(FN + FP)\,(TN + FN)}} \quad \text{Equation 2.48}$$

Kappa Score:

Kappa statistics measures the agreement of prediction with the true class. The true class has a value of 1.0. Kappa formula is illustrated in Equation 2.49.

$$Kappa = \frac{\text{Total accuracy} - \text{Random accuracy}}{1 - \text{Random accuracy}} \quad \text{Equation 2.49}$$

Mean Absolute Error (MAE)

MAE measures the average magnitude of the errors in a set of forecasts, without considering their direction. It measures accuracy for continuous variables. It also calculates the closeness between the predictions to the actual outcome. It is the average of the absolute errors, as illustrated in Equation 2.50

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |Q - Q_i| \quad \text{Equation 2.50}$$

RMSE (Root Mean Square Error)

This is a measure to calculate the values predicted by a model when compared to the actual observed values. It is illustrated in Equation 2.51

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} |Q - Q_i|} \quad \text{Equation 2.51}$$

Anti-malware developers and research community have written so much on the subject matter, as reviewed in related works

2.13.3 Confusion Matrix

The results of the trained model are evaluated to ascertain its generalization and performance with unknown dataset (test dataset). One way to achieve this is the use of confusion matrix. Confusion matrix is a cross table that records the number of occurrences between the predicted and actual classifications. While the columns represent model predictions, rows represent actual values (Kulkarni, 2022; Grandini *et al.*, 2020; Bhandari, 2020). Figure 2.25 depicts confusion matrix of multiclass classification problem for KNN with the following classes (labels): DOS, Probe, R2L and U2R. With this type of confusion matrix, unlike confusion matrix of binary classification problem, parameters such as True positive (TP), True Negative (TN), False Negative (FN), and False Positive (FP) do not apply directly (Markoulidakis *et al.*, 2021; Grandini *et al.*, 2020; Bhandari, 2020).

Because of that, the classes are analyzed one by one, with confusion matrix parameters TP, TN, Probe, R2L and U2R determined in each case. Then performance parameters such as Accuracy, Precision, Recall, F1-score, etc. are computed using appropriate formulae.

		PREDICTED VALUES			
Classes	ACTUAL VALUES	**DOS**	**Probe**	**R2L**	**U2R**
DOS		Cell 1 13979	Cell 3 36	Cell 4 0	Cell 5 0
Normal		Cell 6 355	Cell 8 562	Cell 9 0	Cell 10 0
Probe		Cell 11 114	Cell 13 3213	Cell 14 1	Cell 15 0
R2L		Cell 16 28	Cell 18 10	Cell 19 0	Cell 20 1
U2R		Cell 21 0	Cell 23 0	Cell 24 0	Cell 25 0

Figure 2.25: Multi-class Confusion Matrix for KNN

Algorithm to Compute Multiclass Confusion Matrix Parameters

To calculate TP, TN, FP, and FN for each class, the following observations on Figure 2.23 are taken into consideration (Grandini *et al.,* 2020; Bhandari, 2020). For ease of explanation, the cells are numbered, while the numerical value there-in are generated by the developed python program using scikit learn and other external libraries:

TP: This is the cell value where the predicted and actual value are the same, and for each class, only one TP value is considered

FN: For a class analysis, FN is the sum of values in cells of the corresponding row, except the TP cell value.

FP: The FP value for a class analysis is the sum of cells' values in the corresponding column, except the TP cell value.

TN: In a class analysis, the TN value is the sum of all the values in columns and rows, aside from those in the class being considered.

2.14 Choosing the Right Algorithm in the Design of Intrusion Detection System (IDS)

In machine learning, there is no one algorithm that can solve all problems (Bui et al., 2017; Pham et al., 2018). Several factors affect the choice of machine learning (ML) algorithm to build a model. Business decisions take the center stage when choosing an algorithm to build a model. From technical perspective, there is the need to run all the algorithms on the target dataset and choose the one that gives minimum loss and best available accuracy, this process is called *brute force* process. Common algorithms that feature in IDS design and development include K-NN, Naïve Bayes (NB), Decision tree (DT), GA, among others. (Ravanshad, 2018).

The list is a long one, and given the constrained period of research, and the fact that the researcher may not be exposed to all the tools used to carry out the analysis; other characteristics are considered in the choice of the right set of algorithms, they include (Ravanshad, 2018):

i. **The Type of problem to solve** – Algorithms are designed to solve specific problems using structured data set. Such algorithms may not be adequate in training with unstructured dataset (Ravanshad, 2018). ML algorithms are categorized into supervised learning, unsupervised learning, and reinforcement learning. Supervised learning algorithms are further categorized into classification and regression types. Unsupervised learning is categorized into clustered and outlier types. However, the researcher's interest is in solving a supervised learning problem.

ii. **Size of the training dataset** - High bias and low variance classifiers are good for small training datasets, such algorithms include naïve bayes (NB). They have advantage over low bias and high variance classifiers, like K-NN,

because the latter will likely over fit. However, the low bias and high variance classifiers win in the end, due to incremental learning as the training set grows, because they have low asymptotic error (training error) (Ravanshad, 2018). Since most algorithms may not be good with unstructured dataset, which are usually large, incremental learning is mostly employed.

iii. **Training time** – Different algorithms have different training times, which is a function of size of data and target accuracy. For instance, NB and K-NN train faster than ANN and SVM, which are computationally slow.

iv. **Parameters** – Parameters affect algorithm's behavior, such as error tolerance or number of iterations. It takes trial and error to find a good combination with algorithms having many parameters; although many parameters give greater flexibility to algorithms, accuracy and training time of the algorithm can be sensitive to the right setting (Ravanshad, 2018).

v. **Number of features** – When the number of features in a dataset is very large compared to the observed data points, curse of dimensionality sets in. Indeed, large number of features drags down some learning algorithms.

vi. **Outliers** – Outliers are features of one class that appear in another class, either due to classification error, typographical error, etc., and affect the learning model in training. In this research work, they are considered in the choice of learning algorithms; for instance, both SVM and K-NN ignore outliers in choosing the optimum separable hyper plane and making real-time predictions, respectively (GoodworkLabs, 2018; Harlalka, 2018). Secondly, they use small proportion of the dataset in analysis: support vectors that lie on the maximum margin of the optimal separating plane for

SVM; while K-NN uses only the k value of the nearest neighbors with the shortest distance to determine the class that will house the target object.

Ensemble learning (Bagging) is used to train base classifiers such as LR, KNN, DT, and NB using Random Forest as the learning algorithm. Ensemble learning produces a more efficient classifier, compared to the use of single algorithm, to analyze, detect and classify applications. To justify the combination of these algorithms, the following are their strengths: and shortcomings.

i. K-NN is lazy learning, it loads the entire training data set in memory and waits until the target object is introduced for analysis before it acts.
ii. K-NN selects the class for its target object by computing the shortest distance of selected training elements (the k value) using Euclidean measure; whereas SVM is required to train a model to classify the applications (malware from benign ones).
iii. K-NN is costly when used in higher dimensional spaces, while SVM is more comfortable in higher dimensional spaces.
iv. K-NN consumes memory, while SVM rather conserves memoryKNN is prone to over fitting (being low bias and high variance algorithm), while SVM is not affected by over fitting.

Combining them will correct the over fitting error in KNN.

There are also similarities that make their combination a befitting one:

i. K-NN is scalable and uses incremental learning (Heidari, 2017).
ii. K-NN is biased and uses small section of the training elements as nearest neighbors (Inayat *et al.*, 2016);

iii. K-NN has one optimal hyper parameter, the k value; and uses Euclidean distance measure to compute the shortest distance used to select nearest neighbors.

iv. K-NN is simple and easy to implement

v. Random Forest as a mega algorithm combines the predictions of base models to produce a more realistic model to assess.

Other issues to consider in choosing an algorithm for the development of IDS include (GoodworkLabs, 2018):

i. Business demands

ii. Stakeholders concerns

iii. Awareness of the rules and regulations of e-commerce

2.15 Python Programming Language

Python programming language is a high-level programming language that is easy to read and simple to implement. It is open source, interactive, interpretive, object oriented, and a scripting language. It was created by Guido Van Rossum in 1985 and it has many external libraries such as Scikit-learn, Numpy, Scipy, Pandas, Matplotlib, among others, which are used to extend python functionality, especially in machine learning (Heinold, 2016).

Python is adaptive and can be run in many platforms such as Windows, Macintosh, Unix-Linux systems; Java and dot net (. Net) virtual machines. It is copy righted and its source code is available in http://www.python.org/. It is widely applied. Some Python libraries are briefly discussed:

2.15.1 Scikit-Learn

Scikit-learn is an open-source python library used for machine learning. It can be used in classification, regression, and clustering algorithms in solving Supervised and Unsupervised machine learning problems; such algorithms include support vector machines, random forests, gradient boosting, k-means and DBScan. It is also designed to interoperate with Python numerical and scientific libraries like Numpy, Scipy, Pandas, and Matplotlib.

Scikit-learn is focused on modeling data in collaboration with NumPy and pandas that load, manipulate and summarize data. Some models provided by Skit-learn include (Brownlee, 2014):

i. Clustering – for grouping unlabeled data, such as k-means
ii. Cross-validation – for estimating the performance of supervised models on unseen data.
iii. Datasets – for training and testing models or classifiers to determined their performances.
iv. Dimensionality reduction – for reducing the number of attributes in data: summarization, visualization, and feature selection, such as PCA.
v. Ensemble methods – for combining the predictions of multiple supervised models
vi. Feature extraction – for defining attributes in image and text data.
vii. Feature selection – for identifying meaningful attributes from which to create supervised models.
viii. Parameter tuning – for getting the most out of supervised models
ix. Manifold learning – for summarizing and depicting complex multi-dimensional data.

x. Supervised models – A vast array, not limited to generalized linear models, but includes discriminate analysis, naïve bayes, lazy methods, neural networks, support vector machines, and decision trees.

Scikit_learn also allows users to load data from an external drive or data source. It works on any numeric data stored as NumPy arrays or scipy sparse metrics. To load standard columnar data into a format usable by sciket-learn, use the following (Brownlee, 2014):

NumPy – This is the database structure used for data and model parameters. Input data is presented as NumPy arrays, thus integrating with other python libraries.

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension, NumPy. It provides an object-oriented API (Application Programming Interface) for embedding plots into applications using general-purpose graphics user interface (GUI) toolkits like Tkinter, wxPython, Qt, or GTK+.

Scipy – This algorithm is good at linear algebra, sparse matrix representation, special functions and basic statistical functions

Cython – This language combines C in python and high-level operations.

Pandas library module – It is used for data manipulation and analysis. That is, it offers data structures and operations for manipulating numerical tables and time series. It is free and released under the BSD license. **T**he scikit learn library will be used to implement the proposed system, that not only detect malware but to foil their intrusion in real-time.

Various types of data are used to train and test the models developed; and tools are also found on the internet that are used to evaluate their performances.

2.16 Ethics

Ethics is a method, procedure or perspective for deciding how to act and for analyzing complex problems and issues. Ethics may be rules for distinguishing between right and wrong, such as the golden rule. Ethics may also be defined as norm for conduct that distinguish between acceptable and unacceptable behavior.

While ethical standards govern conduct in medicine, law, engineering and business, ethical norm rather serve the goals and aims of research, and apply to scientific researches and other scholarly or creative activities.

Ethics, therefore, is well funded standards of right and wrong that prescribe what humans ought to do, usually in terms of rights, obligations, benefits to society, fairness or specific virtues. Standards to refrain from include rape, stealing, murder, assault, slander and fraud. Ethical standards enjoin virtues such as honesty, compassion, and loyalty; and right to life, right to freedom from injury and right to privacy (Resnik, 2020).

To adhere to ethical norms in research has many benefits, such as:

i. Norms promote the aims of research such as knowledge, truth, and avoidance of errors. For example, prohibition against plagiarism, falsifying or misrepresenting research data

ii. Ethical standards promote the values that are essential to collaborative work. This includes trust, accountability, mutual respect and fairness. Others include guidelines to authorship, copyright and patenting policies, data sharing policies, and confidentiality rules in peer review.

iii. Ethical standards or norms ensure that researchers can be held accountable to the public. For example, conflict of interest, the human subject's protection and animal care and use.

iv. Ethical norms in research also help to build public support for research. This encourages people to fund research projects, when they trust the quality and integrity of the research.

v. Ethical norms promote moral and social values such as social responsibility, human right, animal welfare, compliance with the law, public health and safety.

2.16.1 Ethical Decision Making in Research

It is important for researchers to learn to interpret, assess and apply various research rules; and make decisions and act ethically in various situations. The following are unethical research practices:

i. Publishing the same paper in two different journals without telling the Editors.

ii. Submitting the same paper to different journals without telling the Editors.

iii. Not informing a collaborator of your intent to file a patent in order to make sure that you are the sole inventor.

iv. Including a colleague as an author on a paper in return for favour even when he or she did not make meaningful contribution to the paper.

v. Failing to keep good research records

vi. Failing to maintain research data for a reasonable period of time.

2.16.2 Ethical Perspectives

At this point, unacceptable trade-offs in health-related issues relating to universal coverage are discussed. Unacceptable choices include low-income countries, to expansion of services covering poor people in society. Universal coverage is currently among efforts to strengthen health systems and include the distribution of health and

health related services. The key issues of fairness and equity that arise on the path of universal health coverage (UHC) include ethically unacceptable trade-offs.

Every person has a right to health in terms of fairness and equity. To realize this, countries must allocate sufficient resources. Fairness is concerned with the overall distribution of benefits and burdens in society. While equity in healthcare is concerned with equitable access to services regardless to socio-economic status. In other words, a fair system will expand service coverage with financial risk protection by giving priority to policies benefiting the worst-off, where worst-off is defined in terms of health and socio-economic status. While in equity, priority to the worst-off is motivated by the right to health.

According to the world health organization (WHO), universal healthcare assumes that all peoples receive quality healthcare services that meet their needs without being exposed to financial hardship in paying for the services (Norheim, 2015). UHC goes beyond clinical and curative services, but include public health, population measures and rehabilitative services. Public health and population measures include campaigns, hygiene, and food safety, vector control and tobacco regulation.

However, in most countries, available resources fall short of what is required to meet all needs. Therefore, priority should be given to services that are likely needed by all. Unfair choices on UHC involves trade-offs between competing goals. A trade-off is a compromise between two or more desirable but competing considerations. Ethical theory is not explicit on the choice of trade-off (Norhein, 2015). Therefore, unacceptable trade-off include:

i. To reduce out-of-pocket (OOP) payments for low and medium priority services before eliminating OOP payments for high priority services. High priority services tend not to benefit the worst-offs such as skilled birth

attendance, easily treatable fatal childhood diseases such as oral dehydration therapy for children with diarrhea and pneumonia.

ii. To first include in the universal coverage scheme only those with the ability to pay, and not include the poor and informal workers, is an unacceptable trade-off. Services should be based on need and not ability to pay.

iii. It is unacceptable to give high priority to every costly service when the health benefits are very small

iv. It is unacceptable to expand coverage for the well-offs before doing so for the worse-off when the cost and benefits are virtually the same. Example, it is unfair to provide health services for tuberculosis detection and treatment in the towns before doing so in the rural villages.

v. It is unacceptable to shift from OOP payment to mandatory pre-payment to boost financing. Other constraints include discrimination based on race, ethnicity, religion, gender, political beliefs and sexual orientation.

2.17 Review of Related Literature

Security of communication systems, data, and transmission channels remains a disturbing issue in recent trends of technological developments. This is because malware developers and hackers employ various means of intrusion using sophisticated tools, most of which are freely available on the internet. More so, cyber threats to critical infrastructure continue unabated, posing a serious national security challenge. Undetected cyber-attacks force users, companies and governments to incur serious financial losses, information loss; and even the reputations of affected industries are destroyed (Sullivan, 2015). To protect and prevent the destruction of security characteristics, anti-malware developers and research community are fighting back using various techniques.

One way to fight the menace of malware and its developers is the use of machine learning techniques.

Electronic Health System

Fan *et al.* (2024) se ek to know why electronic health system is being rebuffed in developing countries like Nigeria. The method used was stimulus-organism-response (SOR) and theory of planned behavior (TPB). The work concluded that fear, education, anxiety and infrastructure accessibility are contributory factors.

Aijaz *et al.* (2023) conducted a systematic literature review on threat modeling and assessment methods in healthcare information technology system. The work opined that with the implanting of medical devices (sensors) in patients, medical healthcare is now seamless, which has made locating, monitoring and treating patients, irrespective of where they may be, common and cheap.

In furtherance to the quest for acceptance of electronic health system, Joukes *et al.* (2019) compared the EHS with the paper-based health record system, using questionnaires. The work discovered that half the people using paper-based version were not satisfied with HER for it did not meet their expectations.

Fatima and Colomo-Palacios. (2018) conducted a systematic mapping review on Health Information System (HIS) security to identify, select, classify and analyze primary studies published in scientific papers. The work observed that electronic healthcare system has become popular targets for ransomware, crypto mining, data theft, and phishing and insider threat because of their high monetary value in the black market.

Olaniyi *et al.* (2015) presented a tele-clinical diagnostic system for effective delivery of medical services to patients in an academic environment. However, the model was not secure enough as it used password-based authentication to control access to the patients' record system, which is weak enough to be breached using brute force method.

Similarly, Tashobya *et al* (2014) conducted assessment on the performance of healthcare systems, especially in low income countries like Nigeria. The objective of this work is tow fold: to develop a set of attributes for good health system performance assessment (HSPA) framework from literature; and to utilize the attributes for a structured approach to learn lessons from international experiences in HSPA.

Electronic Health Record (EHR)

Alobo *et al.* (2020) studied the pioneering work of EHR system to determine health workers perception, challenges, motivation and satisfaction with EHR using structured questionnaire. The results were reduction in transcription cost (88.5%), paper work (97.1%) and administrative cost (91.4%). However, the challenges were

i. Threat to patient privacy (17%)
ii. Poor internet service (65.7%)
iii. Information overload (31.5%)
iv. Power outages (62.9%)

On the whole, health workers were satisfied with electronic Patient Health Records (PHR) for it eased their work. Wesoloski *et al.* (2016) proposed ensemble learning technique using machine learning (ML) tools or models, to monitor keystroke-interface to stop intrusion or unauthorized access to EHR server or computer. However, the work did not mention the type of ensemble learning tools used nor did it mention the data used for the proposed work.

Security

Yeng *et al.* (2021) determined security challenges faced by healthcare workers while performing their duties. The work also attempted to investigate anomaly practices in healthcare system using big data and machine learning techniques. To justify this research, Verizon opined that in 2018 the healthcare sector experienced 503 data

breaches, compromising 15 million records (Yeng *et Al.,* 2021). Unfortunately, more than half of the threats were perpetrated by insiders. The reason for these breaches could be because healthcare records have economic, scientific, and social values, which attracts malicious actors like hackers. It is believed that healthcare records are sold at $1,000 in the black market. However, the ML technique used to analyze these breaches were not mentioned. Another problem is the lack of human firewall, which is information security consciousness in insider staff. That is, access control, which is flexible and subject to abuse by insiders.

Sardi *et al.* (2020) organized a systematic literature review on cyber risk or threat in the healthcare sector. It is unfortunate that not much attention is given to this topic by research community. Indeed, it is research gap for future work. Fortunately, this research work is closing the gap by designing and developing a framework that classifies malware threats using ensemble learning with random forest as the algorithm and DT, SVM, K-NN and NB as base classifiers.

Kumar and Tripathi (2016) modeled threat evaluation for dynamic targets using Bayesian network. The work, however, focused on external aggression and ignored internal attackers (insiders) who may affect the target through sabotage and espionage. Adebisi *et al.* (2015) designed and implemented an automated framework for managing patient's information with the view to reducing inappropriate data, false alarm, wasted time and cost in storage, retrieval and processing patient data. However, no threat model was used to plan security implementation of the system. Shin *et al.* (2014) attempted to prevent information leakage by looking into many security models for health care applications, to ensure security and privacy in electronic health system. To achieve this, the authors employed extended Role-Based Access Control (RBAC) security model. However, the model was not suitable for distributed environment. Also, to protect data

exchange between servers, securehealth architecture based on transport layer security/secure socket layer (TLS/SSL) protocol was implemented. It has the advantage of preventing foreign applications from unauthorized access However, the framework is platform dependent and not scalable (Simplicio *et al.,* 2015).

Cloud Computing Technology

Okediran *et al.* (2022) proposed a cloud based electronic health record framework that is capable of automating storage, retrieval, updating and maintaining patients' medical records in Nigeria. The work used service-oriented architecture (SOA) software development method to derive the framework. However, the work builds the framework from existing infrastructure using routers, servers firewall, gateway, desktop, laptop and smartphones. The authors did not mention the type of software developed by them, and the programming language used to develop the software. In this research work, the framework is designed and developed using ensemble learning tools (bagging) such as RF as training algorithm and base classifiers: SVM, DT, K-NN and NB. Python programming language is used for the development of the framework.

Yeng *et al.* (2020) compared threat modeling methods to determine their suitability for identifying and managing healthcare related threats in cloud computing environment. The work identified threats modeling in pervasive computing to be the best method to use in healthcare security because it can be combined with attack tree (AT), attack graph (AG) and practical thread analysis (PTA) to identify cloud related threat for healthcare. Kushala and Shaylaja (2020) investigated the essential characteristics of cloud computing and multi- cloud computing, their computing difficulties and potential solutions. The work concluded that the shift from on-premises computation to cloud computing technology has not only brought benefit but also introduced new vulnerabilities, not just for end users, but also cloud service providers (CSP). Also,

Mondal *et al.* (2020) discussed the difficulty associated with sharing resources in the cloud as a weakness that should be investigated in future study. It emphasized the difficulties that need addressing to be confidentiality, validity, privacy, cryptography, scalability, and so on.

Syed *et al.* (2020) conducted a review of security threats, procedures and control associated with cloud storage. Other concerns associated with CCT include poor data visibility, storage sinks without protected pointers, enormous data spills, and more. The work provided the following security risks associated with CCT storage to be lack of control, shared servers, data leakage, API storage sinks, and shared data. The work also proposed the following best practices: multi-factor, authentication, data categorization, security encryption and assessment of cloud framework. The work summarized the security risks in CCT to include (Saeed *et al.,* 2022):

i. Account hijacking
ii. Data sanitization
iii. Data control and
iv. Harmful insiders

Azeez and der Vyver (2018) opined that the numerous advantages of using CCT notwithstanding, the full utilization is still being obstructed by security and privacy challenges. The work, therefore, reviewed the various existing literature on mechanisms use to handle security and privacy in electronic healthcare (e-healthcare)

Anand *et al.* (2016) proposed the use of STRIDE-DREAD model to assess threats in cloud-based environment and measure the consequences of their actions. However, the DREAD framework is focused solely on technical threats and does not consider other types of threats, nor does it consider the complexity of an attack and the likelihood of

attacking a particular asset. More so, STRIDE model is complex and not suitable for organizations with limited resources.

Machine learning

Feizollah *et al.* (2014) assessed five ML algorithms (NB, K-NN, DT, MLP and SVM) with 100 malware samples of Android mal-Genome project and 12 samples of benign applications. The work used WEKA ML tools. KNN performed best with TPR of 99.94% against FPR result of 0.06%. However, the training and test data samples were grossly inadequate.

Android Intrusion Detection system (IDS)

Shatnawi *et al.,* (2022) used recursive feature selection (RFS) approach to reduce dimensionality using API calls and permission features. Logistic regression was used as the classifier. However, only one classifier was used, and the data size to use was not mention.

Wang *et al.* (2021) introduced the use of genetic algorithm (GA) to select appropriate features to improve android malware detection (AMD). The ML algorithms used were RF, LR, K-NN and GNB. The best result was obtained from GNB (95.5%) and K-NN (93.3%). However, the data type and size were not mentioned.

Akram *et al.* (2021) proposed an approach for effective intrusion detection system in the electronic healthcare environment to safe guard patient health record (PHR) using adaptive-neuro-fuzzy Inference System (ANFIS). ANFIS is used to resolve uncertainty, reasoning and reducing security threats in network and cloud servers. However, the dataset and type used was not stated or may be the proposal was not implemented.

Cai *et al.* (2020) presented an information gain (IG) based Android malware detection (AMD) framework to select optimum features from extracted features of eight categories. However, the authors did not provide the dataset used, the ML tools used,

nor even the features extracted. Similarly, Singh *et al.* (2020) proposed a framework using latent semantic indexing (LST) to reduce dimensionality of the dataset and improve detection rate. The dataset used was CICInvestAndMal2019, with RF as the classifier scoring 93.92% accuracy. However, the number of records in the dataset used was not provided, and only one classifier was used. In these days of high sophistication of malware evasion techniques, and resistant to detection, one classifier is not good enough, as recommended by the research community.

Garg and Baliyan (2019) attempted to detect zero-day attack vectors using machine learning. The models used were pruning rule-based classification tree (PART), Ripple down rule learner (RIDOR0), SVM, and MLP on a 10-fold cross validation, to improve malware detection accuracy. However, the dataset and size were not mentioned.

Mashiri *et al.* (2017) proposed a framework of malware classification, which analyzed malware dynamically using the concept of information theory and machine learning techniques. The problem with this work is it did not disclose the ML technique used nor did it disclose the data type and size used in the analysis.

Wu and Hung (2014) developed DroidDolphin, a dynamic analysis framework that used GUI, big data and machine learning tools for the detection of malicious applications in Android platform. The data used to train SVM consists of 32,000 benign and malign records, and the test dataset was 3,000 benign applications and 1,000 malign applications consisting of API calls. The results had 86.1% precision and F1-score of 0.875.

However, the dataset to train and test the model was small, more so, the authors used only one classifier to detect malware, when malware has gone so sophisticated that a combination of classifiers is advocated by research community. Hybridization or

ensemble learning methods are recommended. This research work uses ensemble learning (bagging) to train the classifiers.

Lee and Mody (2006) analyzed malware using API call sequences as data; with normalized compression distance (NCD) technique. However, most researches in dynamic analysis use one subset of malware feature to represent its behavior pattern and ignore other ones, generating blind spot, which is exploited by malware. For instance, API based sequences are used in analysis, ignoring network-based sequences, which use packet capture (pcap) files to extract the network flow information.

In this review of related literature, some frameworks presented are contextual with no constraint in ICT resources for a developing country like Nigeria. The techniques used are existing legacy systems or equipment gathered and linked together by heterogeneous healthcare systems. That is, the developed architectures integrated already existing legacy systems into EHR system rather than developing an EHR from scratch using modern ensemble techniques and network-based programming languages like python programming language.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Preamble

Healthcare system in Nigeria is predominantly paper based, and the needed interoperability among healthcare workers across units, departments in a particular hospital, and related hospitals are greatly constrained. The paper-based patient health system is fraught with many problems such as delay in fetching the patient's record, in case of emergency, the insecurity of the files from pests and internal threats, among other problems. These problems, and a lot more have motivated the researcher to design and develop a framework that will identify and report security and privacy issues in electronic healthcare system in Nigeria, which has more advantages than the paper-based healthcare system.

The focus in this research work is to examine the cyber security issues inflicting the electronic health care system, especially in Nigeria, and the safety and protection of patients' records privacy. The dataset used to develop the framework is national science laboratory – knowledge discovery in databases (NSL-KDD) dataset. To effectively use the dataset, it is cleaned, balanced using STROKE, normalized and reduced to an appreciable number using principal component analysis (PCA), and used for the training and testing of machine learning (ML) classifiers to detect and report abnormal applications in the health sector. Cloud computing technology (CCT) is used to train, test and evaluate the classifiers in their ability to detect attack types or threats. These attack (threats) types are grouped into four: DOS, PROBE, U2R and R2L.and identified using the characteristics of normal applications as bench mark.

Ensemble learning technique of ML is used to train classifiers to learn the characteristics of normal applications, and use that knowledge to detect abnormal applications. ML classifiers used in this research work include RF, KNN, DT, NB and LR. Confusion matrix will be used to compare the predictions of the record labels with the expected labels, being a supervised learning problem. The models are further evaluated to determine their performances using the following parameters: accuracy, precision, recall, f1 score and area under the curve.

Related works in literature are compared with the current work to determine their weaknesses in relation to the framework being developed.

3.2 Problem Definition

This research work intends to solve the security and patients' privacy issues in electronic healthcare delivery in Nigeria. Anomaly detection system is used in designing and developing a framework, which will learn the characteristics of normal applications and use that knowledge to detect threats in electronic healthcare system and protect patients' privacy. Machine learning tools such as Random Forest, Decision Tree, K-Nearest Neighbor, Naïve Bayes and Logistic Regression are trained, tested and evaluated for performance using confusion matrix; and performance parameters expected include accuracy, precision, recall, f1 score and area under the curve. The process is carried out in the cloud using the internet as the communication medium. Python is the programming language used to implement the framework.

3.3 Conceptual Framework (Architecture)

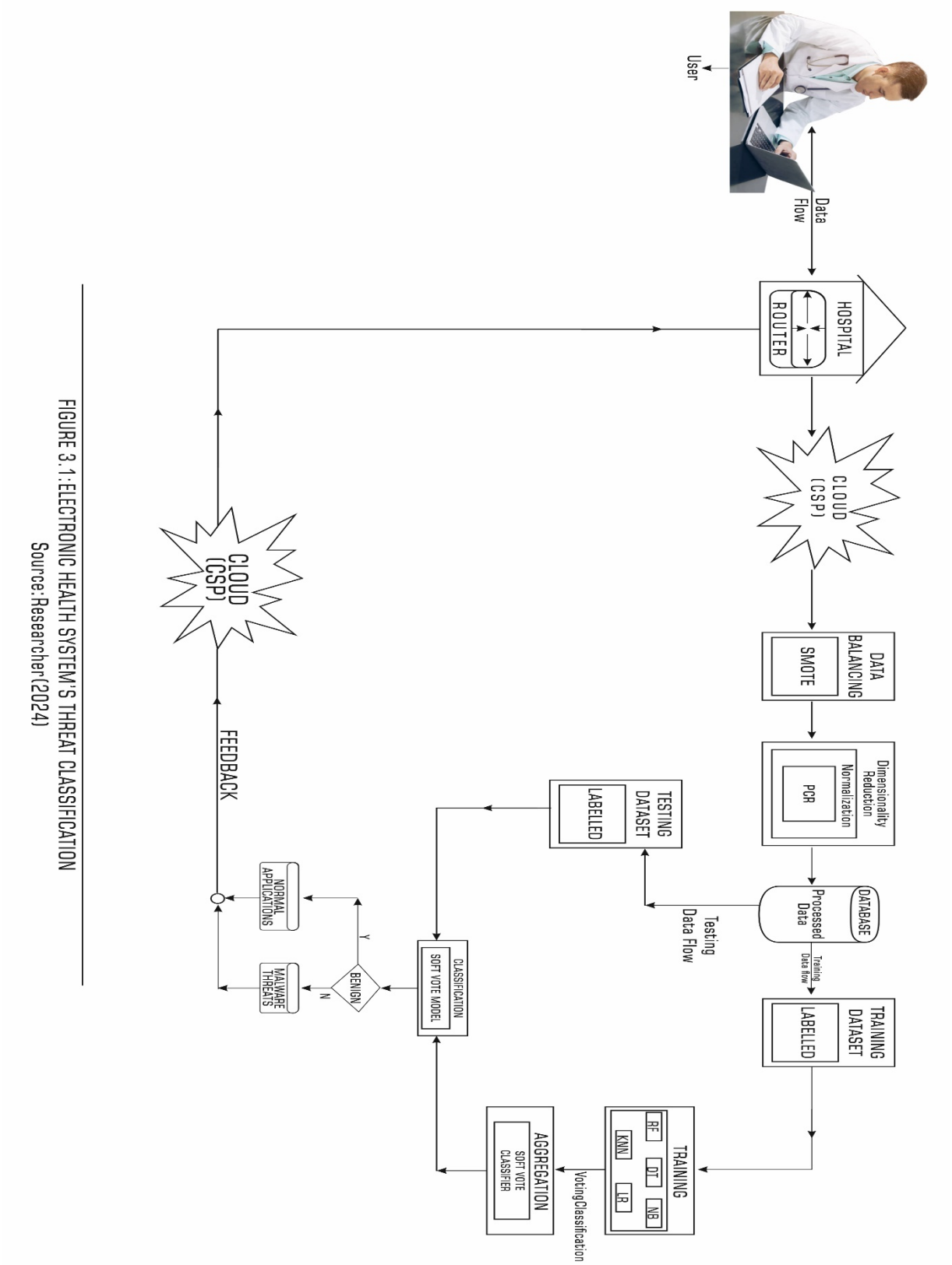


FIGURE 3.1: ELECTRONIC HEALTH SYSTEM'S THREAT CLASSIFICATION
Source: Researcher (2024)

Figure 3.1: Electronic Health System's Threat Classification Framework
Source: Researcher (2024)

Figure 3.1 depicts the system's architecture. The system trains models that analyze applications requests for patients' health records (PHR), either during updating or with the intent of mischieviously infringing on the patient's privacy. It consists of the following components.

i. **The User: -** The user is the human component of this system, who uses either the personal computer, the laptop or mobile device to communicate and access patients' records in healthcare delivery.

ii. **The Hospital -** The hospital reserves the right to control access to the medical records database, which is hoisted in the cloud and managed by cloud service providers (CSP). It receives and stores patients' data obtained from hospital workers, especially medical doctors, laboratory scientists or tecnicians, nurses, pharmacists and even the patients. To reduce its overhead, the data and virtualized device are uploaded to the cloud using mobile host agent, via the proxy device. Figure 3.2 depicts the process of uploading data and vitualized device to the cloud.

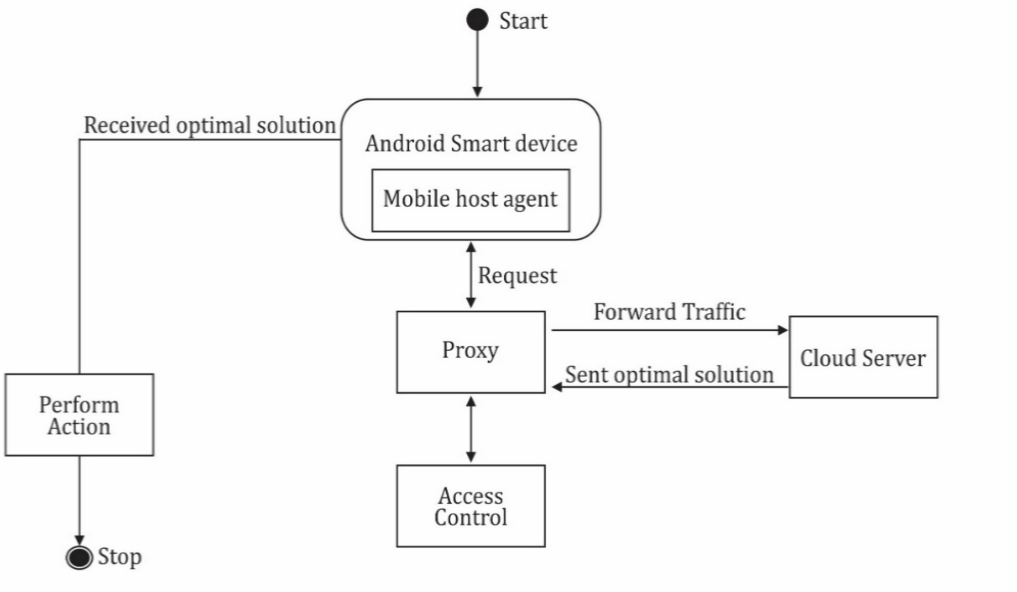


Figure 3.2: Data Movement from the User to the Server in the Cloud

i. **The Cloud**: This is the channel of communication between the hospital staff, the hospital and the patients. It is managed by cloud service providers (CSPs). Another name for the cloud in this field of study is the Internet, which is the platform for data communication to and from the user. The user can be actors such as medical doctors, nurses, laboratory scientists or technicians, pharmacists, the patients, administrators and the suppliers of srvices to the hospital such as insurers and contractors.

ii. **Dataset –** The dataset used in this research work is national science laboratory –knowledge discovery database (NSL-KDD). It consists of 125,973 training data records, and 22,544 test data records. The dataset also consists of 42 features, including labels.

iii. **Data balancing –** The dataset consists of unbalanced data, where some attack types are much more than others. For instance, normal data, neptune (an aspect of DOS) are much more than probe, U2R and R2L labels. To reduce bias in model performance, systematic minority over-sampling technique (SMOTE) from the 'imblearn' library of python, is used to oversample the data types that have insignificant values to balance with the greater ones. After the balance, the data types were reduced to 67,342 records each. The balancing of the record types becomes necessary to anable the training of models that perform well across all data types.

iv. **Dimensionality Reduction –** The dataset used in this research work is preprocessed using normalization, PCA, One_hot_encoder(), and Label_encoder():

    a. **Normalization –** This is the process tthat reduces the data set to a
    b. specific numeric range of say zero to one [0,1].

v. **PCA** further reduces the dataset by combining the original features to form new ones in a much smaller set that will fit the selected training model. It computes eigen values and eigen vectors which are used to determine the principal components (PCs) –accepted features.within a threshold of 95%.

vi. **Model training**: Random Forest algorithm is used to train four base classifiers, such as KNN, NB, DT, and LR. The classifiers predict labels. Voting classifier aggregates the predicted labels into a consensus classifier, called soft vote. The soft vote is used to predict the test dataset and compare with the actual labels using confusion matrix.

vii. **Test data classification**: The test dataset is used to classify abnormal threats from the normal applications, to provide generalization and performance evaluation of the models using the following parameters: accuracy, precision, recall, f1-score and area under the curve. Test dataset serves as proxy to the actual practical dataset, which the model did not train with.. The results of the classification are sent as feedback to the user via the hospital.

viii. **Normal Application** – These are applications from the test dataset, which have the characteristics of normal or accepted data labels. At the detection of this application, the user is notified via the hospital.

ix. **Malware Threats** – these are applications from the test dataset, whose characteristics deviate from the norms of normal applications. They are classified as threats to the system and are mischievious. Once the anomalous application is identified, the report is sent to the user, through the hospital.

Figure 3.3 depicts the flow chat representing the various stages of data processing in the architecture (Figure 3.1).

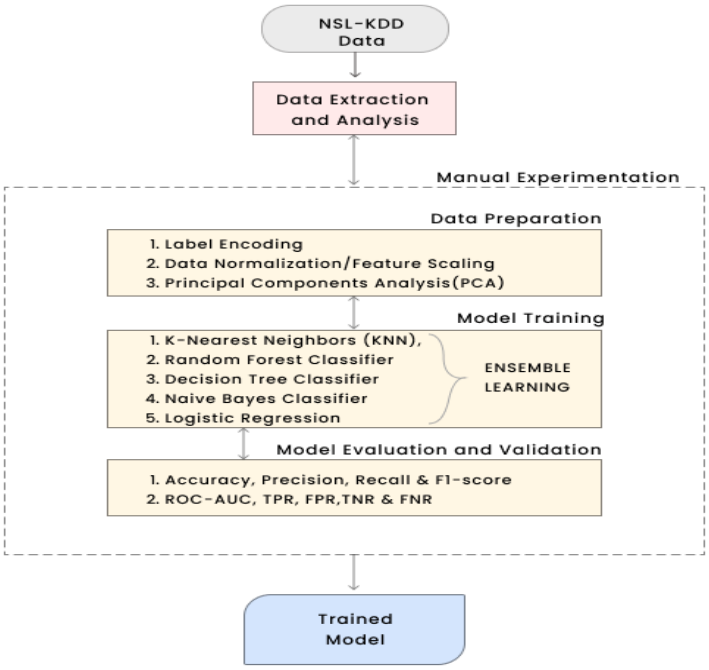


Figure 3.3: Flow Chart Showing the Stages of Data Processing in Figure 3.1

3.4 Framework Development Tools/Algorithms

The laptop system used has the following configurations: Operating System: Windows 10 Pro, with 64-bit word length, CPU: Intel ® Celeron® 1000M, 1.80GHZ; RAM: 4.00 GB, HDD 500 GB, DVD Drive, Keyboard, and Mouse.

Printer: hp LaserJet p2035

Applications: Microsoft Office Suite ver. 16: MS Word, Excel, & Power Point

Programming language used: Python 3.8 and its external libraries (Scikit learn, Pandas, Numpy, matplotlib.pyplot, etc.)

Environment used is: Cloud Computing Technology

3.4.1 Ensemble (Bagging) Learning Classifiers Used

IDS is designed and developed using the following bagging classifiers:

i. **Random Forest** – Is bagging training algorithm; for classification, the output of RF is the class selected by most trees. It corrects over fitting of the training set.

ii. **K-Nearest Neighbor (K-NN)** – Base model 1; measures the distance of each Training element in memory to the object data using Euclidean measure; and assigns the test object to the class with majority vote.

iii. **Logistic Regression** – This is supervised machine learning algorithm that accomplishes binary classification tasks by predicting the probability of an outcome. Its outcomes are limited to two possibilities, yes/no, true/ false, 1/ 0.

iv. **Decision Tree**: These are tree-based models that learn hierarchical decision rule from training data, they are used for both classification and regression problems.

v. **Naïve Bayes (NB)** – This is a probabilistic classifier based on Bayes' theorem, which assumes that the features are conditionally independent given the class labels.

vi. **Voting technique** (Soft vote).- This is a tool (function) in scikit learn module of Python. It groups the predictions of the base models; determines a consensus classifier using soft-vote technique (probabilistic vote), and uses that classifier to classify test dataset.

vii. **Output:** These are classified normal or abnormal records and their labels, grouped into Normal, DOS, Probe, U2R and R2L. The results are sent back to the user as feedback, via the hospital.

3.5 Data Collection

To identify threats or attacks ravaging the electronic health system, the dataset used to design and develop the framework is NSL-KDD dataset, obtained from Kaggle - a public data repository (Github, 2020). It is presented in the form of Attribute Relation File Format (ARFF). ARFF file is an ASCII text file that describes the list of instances that shares a set of attributes. It is divided into two subsections: Header subsection and Data subsection. The Header describes the features of the dataset, while the Data section describes the observations of the dataset as depicted in Figure 3.4.

```
%       The Header
%       Title:      NSL-KDD Data set
%       Source:     GitHub Inc. (2020)
%
%
@RELATION       NSL-KDD Data set

@ATTRIBUTE      “duration”                      real
@ATTRIBUTE      “protocol_type”                 {tcp, udp, icmp}

@ATTRIBUTE      “service”                       {aol, auth, bgp, courier
…}
@ATTRIBUTE      “flag”                          {OTH, REJ, RSTO, RSTR
...}
@ATTRIBUTE      “src_bytes”                     real
@ATTRIBUTE      “dst_bytes”                     real
@ATTRIBUTE      “land”                          {‘0’, ‘1’}
@ATTRIBUTE      “wrong_fragment”                real
@ATTRIBUTE      “urgent”                        real
@ATTRIBUTE      “hot”                           real
@ATTRIBUTE      “num_failed_logins”             real
@ATTRIBUTE      “logged_in”                     {‘0’, ‘1’}
@ATTRIBUTE      “num_compromised”               real
@ATTRIBUTE      “root_shell”                    real
@ATTRIBUTE      “su_attempted”                  real
@ATTRIBUTE      “num_root”                      real
@ATTRIBUTE      “num_file_creations”            real
@ATTRIBUTE      “num_shells”                    real
@ATTRIBUTE      “num_access_files”              real
@ATTRIBUTE      “num_outbound_cmds”             real
@ATTRIBUTE      “is_host_login”                 {‘0’, ‘1’}
@ATTRIBUTE      “is_guest_login”                {‘0’, ‘1’}
@ATTRIBUTE      “count”                         real
```

@ATTRIBUTE “srv_count” real
@ATTRIBUTE “serror_rate” real
@ATTRIBUTE “srv_serror_rate” real
@ATTRIBUTE “rerror_rate” real
@ATTRIBUTE “srv_rerror_rate” real
@ATTRIBUTE “same_srv_rate” real
@ATTRIBUTE “diff_srv_rate” real
@ATTRIBUTE “srv_diff_host_rate” real
@ATTRIBUTE “dst_host_count” real
@ATTRIBUTE “dst_host_srv_count” real
@ATTRIBUTE “dst_host_same_srv_rate” real
@ATTRIBUTE “dst_host_diff_srv_rate” real
@ATTRIBUTE “dst_host_same_src_port-rate” real
@ATTRIBUTE “dst_host_srv_diff_host_rate real
@ATTRIBUTE “dst_host_serror_rate” real
@ATTRIBUTE “dst_host_srv_serror_rate” real
@ATTRIBUTE “dst_host_rerror_rate” real
@ATTRIBUTE “dst_host_srv_rerror_rate” real
@ATTRIBUTE “class” {‘normal’, ‘anomaly’}

% *The Body of ARFF*

@DATA

0,tcp,http,SF,209,442,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,26,26,0,0,0,0,1,0,0,235,255,1,0,0,0.02,0,0,0,0,normal,21

5,tcp,pop_3,SF,26,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,233,214,0.92,0.01,0,0,0,0,0.04,0,guess_passwd

0,tcp,http,SF,327,294,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,9,10,0,0,0,0,1,0,0.2,255,255,1,0,0,0,0,0,0,normal,21

4,tcp,pop_3,SF,32,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,161,0.63,0.02,0,0,0,0,0.13,0,guess_passwd,15

0,udp,private,SF,52,54,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0,0,0,0,1,0,0,255,255,1,0,0.87,0,0,0,0,0,normal,17

0,tcp,http,SF,295,527,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,8,8,0,0,0,0,1,0,0,124,255,1,0,0.01,0.03,0,0,0,0,normal,21

0,tcp,http,SF,316,356,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,4,4,0,0,0,0,1,0,0,38,255,1,0,0.03,0.02,0,0,0,0,normal,

0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,10,10,0,0,0,0,1,0,0,35,10,0.29,0.11,0.29,0,0,0,0,0,teardrop,11

3,tcp,pop_3,SF,30,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,247,0.97,0.01,0,0,0,0,0.02,0,guess_passwd,18

1,tcp,telnet,RSTO,123,178,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,1,1,0,0,255,12,0.05,0.01,0,0,0,0,0.03,0.67,guess_passwd,13

0,tcp,http,SF,311,294,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,27,29,0,0,0,0,1,0,0.1,255,255,1,0,0,0,0,0,0,0,normal,21

0,tcp,http,SF,227,406,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,34,0,0,0,0,1,0,0.09,64,255,1,0,0.02,0.03,0,0,0,0,normal,21

0,udp,private,SF,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,5,5,0,0,0,0,1,0,0,255,253,0.99,0.01,0,0,0,0,0,0,snmpguess,13

0,tcp,http,SF,314,358,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,10,0,0,0,0,1,0,0,231,255,1,0,0,0.02,0,0,0,0,normal,21

0,tcp,http,SF,243,269,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,21,21,0,0,0,0,1,0,0,255,255,1,0,0,0,0,0,0,0,normal,21
0,tcp,ftp_data,SF,19,00,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,122,15,0.04,0.05,0.04,0.13,0,0,0,0,normal,21

210,tcp,telnet,SF,126,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,248,0.97,0.01,0,0,0,0,0.02,0.02,guess_passwd,16

0,tcp,http,SF,203,454,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,11,0,0,0,0,1,0,0.18,255,255,1,0,0,0,0,0,0,0,normal,21

0,tcp,finger,SF,5,381,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,2,0.01,0.03,0,0,0.02,0,0.7,0,normal,16

0,tcp,http,SF,279,526,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,13,0,0,0,0,1,0,0.15,2,255,1,0,0.5,0.02,0,0,0,0,normal,21

0,udp,private,SF,42,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,255,253,0.99,0.01,0.01,0,0,0,0,0,snmpguess,14

1,tcp,telnet,SF,24,715,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0.67,0.33,0.33,1,0.67,255,67,0.26,0.02,0,0,0,0.01,0.7,0.69,mscan,11

0,udp,domain_u,SF,46,86,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,248,246,0,0,0,0,0.99,0.01,0,255,254,1,0.01,0,0,0,0,0,0,normal,18

4,tcp,pop_3,SF,28,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,90,77,0.84,0.03,0.01,0.03,0,0,0.11,0,guess_passwd,7

4,tcp,pop_3,SF,25,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,70,67,0.94,0.03,0.01,0.03,0,0,0,0,guess_passwd,6

0,udp,private,SF,49,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,4,0,0,0,0,1,0,0,255,254,1,0.01,0.01,0,0,0,0,0,snmpguess,16

2,tcp,imap4,RSTO,0,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,6,0,0,0.67,1,0.33,1,1,5
1,54,0.27,0.1,0.02,0.04,0,0,0.92,0.26,mscan,12

0,tcp,telnet,SF,125,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,109,
0.43,0.02,0,0,0,0,0.01,0.03,guess_passwd,10

4,tcp,pop_3,SF,31,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,208,0.
82,0.02,0,0,0,0,0.07,0,guess_passwd,18

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,27,0,0,0,0,1,0,1,2,60,1,0,1,
0.5,0,0,0,0,ipsweep,17

0,udp,domain_u,SF,46,77,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,188,188,0,0,0,0,1,0,0,2
55,255,1,0,0.01,0,0,0,0,0,normal,18

0,udp,domain_u,SF,45,82,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,159,160,0,0,0,0,0.99,0.
01,0.02,255,250,0.98,0.01,0,0,0,0,0,0,normal,18

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,21,0,0,0,0,1,0,1,2,42,1,0,1,
0.26,0,0,0,0,nmap,16

0,tcp,ftp_data,SF,43,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,255,62,0.
24,0.02,0.01,0,0.87,0.97,0,0,normal,14 *

0,udp,private,SF,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,156,146,0,0,0,0,0.94,0.01,0,
255,208,0.82,0.03,1,0,0,0,0,0,satan

0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,5,17,1,0,
1,0.12,0,0,0,0,warezclient,13

0,tcp,finger,SF,9,196,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,5,5,0,0,0,0,1,0,0,155,7,0.05,
0.06,0.01,0,0.01,0,0,0,satan,0

0,icmp,ecr_i,SF,520,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,184,184,0,0,0,0,1,0,0,255,
255,1,0,1,0,0,0,0,0,smurf,18

0,icmp,eco_i,SF,20,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,3,0.01
,0.02,0.01,0,0,0,0,0,satan,6

1,tcp,ftp,RSTO,0,133,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,1,1,0,0,2,2,0.5,1,0
.5,1,0,0,1,1,ipsweep,9

0,udp,domain_u,SF,34,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,0,0,0,0,1,0,1,213,22,
0.1,0.03,0.1,0,0,0,0,0,normal,21

0,icmp,ecr_i,SF,1480,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,1,25,1,0,
1,0.52,0,0,0,0,pod,17

0,tcp,pop_3,RSTO,0,36,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0,0,1,1,0.33,1,1,189,
60,0.24,0.03,0.01,0.03,0,0,0.88,0.98,mscan,15

0,tcp,ftp,SF,26,157,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,1,0,0,155,71,0.46,
0.03,0.01,0,0,0,0,0,guess_passwd,7

0,tcp,telnet,S3,0,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,1,0,0,255,117,0.4
6,0.02,0,0,0.53,0.37,0.01,0.01,processtable,18

0,udp,private,SF,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,254,1,
0.01,0,0,0,0,0,0,snmpguess,12

0,tcp,telnet,RSTO,124,188,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,1,1,0,0,255,
254,1,0.01,0,0,0.01,0.01,0.02,0.02,guess_passwd,10

8169,tcp,telnet,SF,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,19,0
.07,0.82,0,0,0,0,0.8,0,processtable,12

0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,192,19
1,0.99,0.01,0.01,0,0,0,0,0,snmpgetattack,2

1,tcp,telnet,RSTO,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,7,0,0,1,0.57,1,0,0.57,96
,87,0.34,0.05,0.01,0.02,0,0,0.9,0.7,mscan,11

0,tcp,pop_3,SF,30,217,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,2,0.0
1,0.02,0,0,0,0,0,0,guess_passwd,4

0,udp,other,SF,23,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,1,0,0.03
,0,0,0,0,0,0,multihop,9

0,tcp,telnet,SF,123,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,255,49,0
.19,0.02,0,0,0,0,0,0.01,0.04,guess_passwd,11

0,udp,private,SF,105,147,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,237,23
5,0.99,0.01,0,0,0,0,0,0,snmpgetattack,1

,tcp,telnet,SF,6,54,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,5,0,0.2,0.5,0.4,0.5,0.75,0.6,2
55,83,0.33,0.1,0,0,0,0,0.33,0.48,mscan,11

0,udp,private,SF,47,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,255,1,
0,0.01,0,0,0,0,0,snmpguess,15

0,tcp,telnet,SF,122,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,25,0
.1,0.02,0,0,0,0,0,0.04,guess_passwd,9

0,tcp,telnet,SF,120,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0,255,115,
0.45,0.02,0,0,0,0,0.01,0.03,guess_passwd,11

0,icmp,ecr_i,SF,508,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,182,182,0,0,0,0,1,0,0,255,
255,1,0,1,0,0,0,0,0,smurf,18

1,tcp,smtp,SF,2599,293,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0,0,1,0,0,255,138
,0.54,0.15,0,0,0,0,0.44,0,mailbomb,11

0,udp,other,SF,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,140,1,0,0,0,0,0.01,0.01,0,8,2,0
.25,0.62,0.25,0,0,0,0.12,0,satan,10

0,icmp,eco_i,SF,20,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,55,0,0,0,0,1,0,1,3,47,1,0,1,0.26,0,0,0,0,saint,15

0,udp,private,SF,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0,0,1,0,0,255,255,1,0,0.02,0,0,0,0,snmpguess,16

8,tcp,ftp,SF,220,688,0,0,0,4,0,1,0,0,0,0,4,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,53,27,0.51,0.08,0.02,0.00,0.00,0.00,0.00,0.00,buffer_overflow,1

0,udp,private,SF,47,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,15

0,tcp,ftp_data,SF,12,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,7,25,1.00,0.00,1.00,0.12,0.00,0.00,0.00,0.00,warezmaster,12

0,tcp,ftp,SF,26,157,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,129,0.51,0.03,0.00,0.00,0.00,0.00,0.00,0.00,guess_passwd,9

0,udp,private,SF,46,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,6,6,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,snmpguess,15

282,tcp,ftp,SF,163,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,5,0.02,0.24,0.00,0.00,0.34,0.00,0.24,0.00,warezmaster,15

0,udp,other,SF,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,145,1,0.00,0.00,0.00,0.00,0.01,1.00,0.00,255,1,0.00,0.62,0.99,0.00,0.00,0.00,0.00,0.00,satan,19

282,tcp,ftp,SF,161,597,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,97,0.38,0.03,0.00,0.00,0.00,0.00,0.00,0.00,warezmaster,13

282,tcp,ftp,SF,164,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,31,0.12,0.03,0.00,0.00,0.40,0.00,0.02,0.00,warezmaster,15

,tcp,ftp,SF,162,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,17,0.07,0.02,0.00,0.00,0.14,0.00,0.25,0.00,warezmaster,12

282,tcp,ftp,SF,162,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,57,0.22,0.03,0.00,0.00,0.13,0.00,0.02,0.00,warezmaster,11

1179,udp,other,SF,146,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,33,14,0.42,0.12,1.00,0.00,0.00,0.00,0.00,0.00,snmpgetattack,0

0,udp,private,SF,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,snmpgetattack,1

0,udp,private,SF,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,5,5,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,15

0,icmp,ecr_i,SF,1008,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,303,303,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,smurf,18

282,tcp,ftp,SF,158,597,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,72,0.28,0.03,0.00,0.00,0.05,0.00,0.02,0.00,warezmaster,12

0,udp,private,SF,49,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,255,253,0.99,0.01,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,14

0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,
1.00,0.00,0.00,255,253,0.99,0.01,0.01,0.00,0.00,0.00,0.00,0.00,snmpgetattack,
3

4,tcp,pop_3,SF,28,103,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,255,214,0.84,0.02,0.00,0.00,0.00,0.00,0.04,0.01,guess_passwd,15

282,tcp,ftp,SF,160,597,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,255,50,0.20,0.02,0.00,0.00,0.00,0.00,0.55,0.00,warezmaster,14

,udp,private,SF,41,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,255,254,1.00,0.01,0.01,0.00,0.00,0.00,0.00,0.00,snmpguess,12

0,icmp,eco_i,SF,30,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,247,247,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,normal,20

0,icmp,eco_i,SF,30,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,227,227,1,
0,1,0,0,0,0,0,normal,20

0,udp,domain_u,SF,35,91,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,4,0,0,0,0,1,0,0,75,25
5,249,0.98,0.01,0.01,0,0,0,0,0,normal,21

0,udp,domain_u,SF,22,75,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,1,0,0,0,0,0.5,1,0,62,5
9,0.950.03,0.02,0,0,0,0,0,normal,18

Figure 3.4: Input Dataset in Attribute-Relation File Format (ARFF)
Source: GitHub Inc (2020)

From Figure 3.4 ARFF consists of twenty-three (23) data classes. Some data classes do not have enough values or records to make them fit to train with. To fix this problem, the 23 class variables are grouped into normal type and four categories of attack. This is depicted in Table 3.1 and Figure 3.5

Table 3.1: Number of Records in each Class of the Target Variable (labels)

Classes	Number of records
Normal	67343
neptune	41214
satan	3633
ipsweep	3599
Portsweep	2931
smurf	2 646
nmap	1493
back	956
teardrop	892
warezclient	890
pod	201
guess_passwd	53
buffer_overflow	30
Warezmaster	20
land	18
imap	11
rootkit	10
loadmodule	9
ftp_write	8
multihop	7
phf	4
perl	3
spy	2

In Figure 3.5, a bar chart depicts the unbalanced attack types and the pie chart depicts when these labels are grouped into normal and four attack types, DOS, PROBE, U2R and R2L

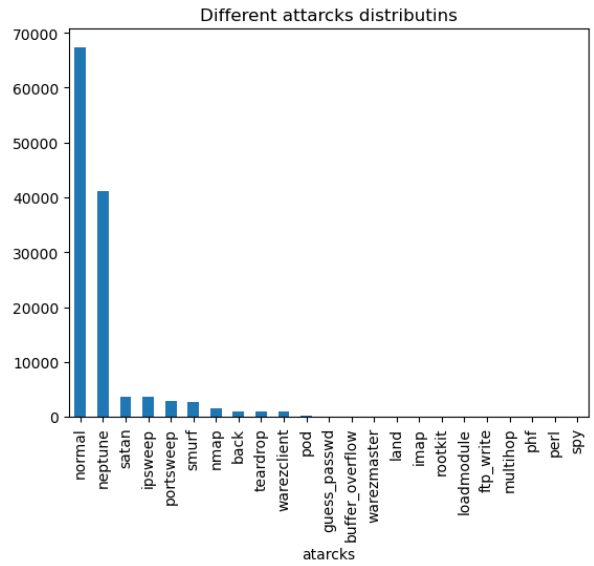
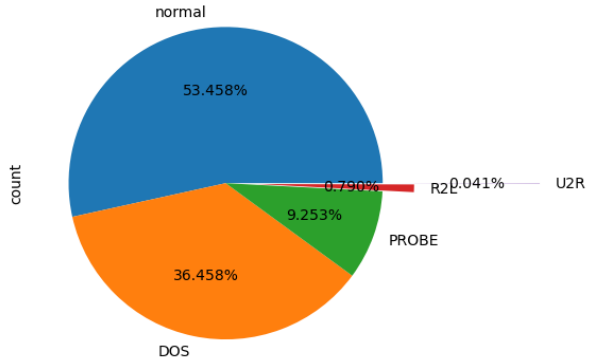
Ungrouped attacks types	Grouped attacks types using the above table
	

Figure 3.5: Bar Chart Depicting Imbalanced Attack Types, and a Pie Chart Depicting the Groupings of these Labels into Normal, DOS, PROBE, U2R and R2L Attack Types

The records of Table 3.1 are further rearranged into the four groups as depicted in Table 3.2

Table 3.2: The Grouping of NSL-KDD Attack Types

ATTACK CLASS	ATTACK NAMES
DOS	Back, Land, Neptune, Pod, Smurf, Teardrop, Apache2, Udpstorm, Processtable, Worm.
Probe	Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint
R2L	Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named
U2R	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps

Source: Researcher (2024)

The data subsection of Figure 3.4 is in comma separated version (CSV) and it is classified into symbolic features, which are four (4): protocol type, services, flag, and class label; fifteen (15) float data type; nineteen (19) integer data type; and four (4) binary type , {0, 1} (Github Inc, 2020). The dataset consist of 125, 973 records (rows) in the training dataset; and 22, 544 records (rows) in the test dataset. Each record has forty two (42) columns (or attributes or criteria, or features or dimensions).

Each data record ends with the class label (type), which is either normal or anomalous (malware). From Figure 3.4, symbolic features determine the state of an application (normal or malware). Malware have affinity for icmp protocol, while normal applications use tcp or udp; Service: The following services are used by malware: private, mtp, finger, netbios_dgm, supdup, uucep_path, ftp_data, z39_50, and csnet_ns. Normal applications use the following services: http, private, smtp, ftp_data, other, telnet, and domain_u. Flag: Malware usually use S0 and REJ flags; while normal apps use SF (safe flag) flag. However, if the protocol used by an application is icmp and the

service is eco_i then the application is ipsweep malware (Probe). If the protocol used by an app is tcp and flag used is S0 or REJ, then such an app is Neptune (DOS) malware. Table 3.3 depicts the description of the forty three features in NSL-KDD dataset.

Table 3.3: Description of NSL-KDD Dataset Features

S/N	FEATURE NAME	DESCRIPTION	TYPE	VALUE TYPE
1	Duration	Length of time duration of the connection	Continuous	Integers
2	Protocol Type	Protocol used in the connection	Categorical	
3	Service	Destination network service used	Categorical	
4	Flag	Status of the connection – Normal or Error	Categorical	
5	Src Bytes	Number of data bytes transferred from source to destination in single connection	Continuous	Integers
6	Dst Bytes	Number of data bytes transferred from destination to source in single connection	Continuous	Integers
7	Land	If source and destination IP addresses and port numbers are equal then, this variable takes value 1 else 0	Binary	{ 0 , 1 }
8	Wrong Fragment	Total number of wrong fragments in this connection	Discrete	{ 0,1,3 }
9	Urgent	Number of urgent packets in this connection. Urgent packets are packets with the urgent bit activated	Discrete	Integers
10	Hot	Number of “hot“ indicators in the content such as: entering a system directory, creating programs and executing programs	Continuous	Integers
11	Num Failed Logins	Count of failed login attempts	Continuous	Integers
12	Logged In	Login Status : 1 if successfully logged in; 0 otherwise	Binary	Integers
13	Num Compromised	Number of "compromised” conditions	Continuous	Integers
14	Root Shell	1 if root shell is obtained; 0 otherwise	Binary	{ 0 , 1 }
15	Su Attempted	1 if "su root" command attempted or used; 0 otherwise	Discrete (Dataset contains ‘2’ value)	Integers
16	Num Root	Number of "root" accesses or number of operations performed as a root in the connection	Continuous	Integers
17	Num File Creations	Number of file creation operations in the connection	Continuous	Integers
18	Num Shells	Number of shell prompts	Continuous	Integers
19	Num Access Files	Number of operations on access control files	Continuous	Integers
20	Num Outbound Cmds	Number of outbound commands in an ftp session	Continuous	Integers
21	Is Hot Logins	1 if the login belongs to the "hot" list i.e., root or admin; else 0	Binary	{ 0 , 1 }

22	Is Guest Login	1 if the login is a "guest" login; 0 otherwise	Binary	{ 0 , 1 }
23	Count	Number of connections to the same destination host as the current connection in the past two seconds	Discrete	Integers
24	Srv Count	Number of connections to the same service (port number) as the current connection in the past two seconds	Discrete	Integers
25	Serror Rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in count (23)	Discrete	Floats (hundredths of a decimal)
26	Srv Serror Rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in srv_count (24)	Discrete	Floats (hundredths of a decimal)
27	Rerror Rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in count (23)	Discrete	Floats (hundredths of a decimal)
28	Srv Rerror Rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in srv_count (24)	Discrete	Floats (hundredths of a decimal)
29	Same Srv Rate	The percentage of connections that were to the same service, among the connections aggregated in count (23)	Discrete	Floats (hundredths of a decimal)
30	Diff Srv Rate	The percentage of connections that were to different services, among the connections aggregated in count (23)	Discrete	Floats (hundredths of a decimal)
31	Srv Diff Host Rate	The percentage of connections that were to different destination machines among the connections aggregated in srv_count (24)	Discrete	Floats (hundredths of a decimal)
32	Dst Host Count	Number of connections having the same destination host IP address	Discrete	Integers
33	Dst Host Srv Count	Number of connections having the same port number	Discrete	Integers
34	Dst Host Same Srv Rate	The percentage of connections that were to different services, among the connections aggregated in dst_host_count (32)	Discrete	Floats (hundredths of a decimal)
35	Dst Host Diff Srv Rate	The percentage of connections that were to different services, among the connections aggregated in dst_host_count (32)	Discrete	Floats (hundredths of a decimal)
36	Dst Host Same Src Port Rate	The percentage of connections that were to the same source port, among the connections aggregated in dst_host_srv_count (33)	Discrete	Floats (hundredths of a decimal)
37	Dst Host Srv Diff Host Rate	The percentage of connections that were to different destination machines, among the connections aggregated in dst_host_srv_count (33)	Discrete	Floats (hundredths of a decimal)
38	Dst Host Serror Rate	The percentage of connections that have activated the flag (4) s0, s1, s2	Discrete	Floats (hundredths

		or s3, among the connections aggregated in dst_host_count (32)		of a decimal)
39	Dst Host Srv Serror Rate	The percent of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in dst_host_srv_count (33)	Discrete	Floats (hundredths of a decimal)
40	Dst Host Rerror Rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_count (32)	Discrete	Floats (hundredths of a decimal)
41	Dst Host Srv Rerror Rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_srv_count (33)	Discrete	Floats (hundredths of a decimal)
42	Class	Classification of the traffic input	Categorical	Strings
43	Difficulty Level	Difficulty level	Discrete	Integers

Source: Saperito (2019)

From Table 3.3, there are forty-three features distributed as follows:

i. Input features are 41

ii. Label feature is 1

iii. Score or Severity of the traffic input is 1

The features are categorized into four:

i. Intrinsic features, 1 – 9

ii. Content features, 10 – 22

iii. Time_based features, 23 – 31

iv. Host_based features, 32 - 41

Intrinsic features can be derived from the packet header, without looking into the payload. They hold the basic information about the packet. The category contains features 1 - 9.

Content features hold information about the original packets, they are sent in multiple pieces rather than one. With this information, the system can access the payload. This category contains features 10 – 22.

Time-based features hold the analysis of the traffic input over a two-second window and contains information like “how many connections it attempted to make to the same host”. These features are mostly counts and rates rather than information about the contents of the traffic. The category contains features 23 – 31.

Host_based features are similar to time-based features, except that instead of analyzing over a two –second window, it analyzes over a series of connections made (how many requests made to the same host over x-number of connections). The features are designed to access attacks, which spans longer than a 2-second window time span. This category contains features 32 – 41.

The feature types can be further broken into:

i. Categorical features are four (2, 3, 4, 42)

ii. Binary features are six (7, 12, 14, 20, 21, and 22)

iii. Discrete features are twenty-two (8, 9, 15, 23 – 41, and 43)

iv. Continuous features are ten in number (1, 5, 6, 10, 11, 13, 16, 17, 18 and 19)

Protocol and service types describe the connection, while flag type describe the status of the connection, and whether a flag was raised or not. Each value in flag represents a status the connection had. Table 3.4 depicts the status of each flag in a connection.

Table 3.4: Status of Each Flag in NSL-KDD Dataset

S/N	Flag	Description
1	SF	Normal establishment and termination. It is the same as State S1. However, for S1 there will be no byte count in the summary, while SF has byte counts.
2	REJ	Connection attempt rejected
3	S0	Connection attempt seen, no reply
4	S1	Connection established, not terminated
5	S2	Connection established and close attempt by originator seen (but no reply from the responder)
6	S3	Connection established and close attempt by responder seen (but no reply from the originator

7	RSTO	Connection reset by the originator
8	RSTR	Connection reset by the responder
9	OTH	No SYN seen, just midstream traffic (a partial connection that was not later closed)
10	RSTOSO	Originator sends a SYN followed by a RST, no SYN-ACk from the responder.
11	SH	Originator sent a SYN followed by a FIN, no SYN-ACK from the responder (hence the connection was half open)
12	SHR	Responder sent a SYN- ACK followed by a FIN, however, no SYN from the originator. (This flag is not in NSL-KDD dataset)

Source: Saperito (2019); SYN: Synchronization, ACK: Acknowledgement; RST: Reset, to indicate that the originator will neither accept nor send data; FIN: Specifies that no more data will be sent by the sender

3.5.1 Corroborating NSL-KDD Dataset Using Significant Features in ARFF File

Some columns or features in ARFF file are more significant than others. For instance, ambiguities are found in some records with same values for all the forty-one features in both training and test datasets. Such redundant and ambiguous features are sifted out during dimensionality reduction using PCA preprocessing. The significant features are depicted in Table 3.5

Table 3.5: Significant Features

S/N	Attribute Names	Description
1	Src_bytes	This is the number of data bytes that is transferred from source to destination in a single connection.
2	Count	This is the number of connections to the same destination host, as the current connection in the past two seconds
3	Srv_count	This is the number of connections to the same service (port number), as the current connection in the past two seconds.
4	Logged_in	This indicates the log in status: 1 if successfully logged in, and 0 otherwise.
5	Num_compromised	This is the number of compromised conditions, especially network systems infected and turned into zombies, then added into the botnet family. Another example is the "SQL injection" where malicious code is embedded into a normal application to corrupt it.
6	Dst_host_count	This is the number of connections having the same destination host IP address
7	Num_outbound_cmd	This is the number of outbound commands issued by command and control (C&C) servers in a file transfer (ftp) protocol session to steal data, customer contact list, and carry out nefarious acts like SEND_SMS, READ_CONTACT, WRITE_CALL_LOG.

Source: Github Inc. (2020)

All the attributes in @ATTRIBUTE statement of ARFF file are open to both normal applications and malware, and each record in the @DATA subsection is a defined data type (normal or any of the attack types).

The significant features are used to corroborate the dataset. The models used to validate the dataset include min-max normalization, PCA process, and KNN. The following were computed:

i. Computation of compromised systems using smurf (DOS), ipsweep (probe), warezclient (R2L) and buffer_overflow (U2R) attack types. The number of systems compromised within two seconds attack time ranged from 0.76 to 0.99. This burtressed the min-max normalization technique that reduced data to a range of [0,1] for better performance of the trained models.

ii. Computation of principal components used to reduce the dimensions of the training dataset. This was achieved by using normal data type and ipsweep, a probe attack type, to build a covariance matrix in two dimensions. The features used in the computation were “count” and “srv_count” both were integer types. “count” indicates the number of connections to the destination host; and “srv_count” indicates the number of connections to the port numbers. This process highlights how Eigen vectors and Eigen values were computed and used to derive principal components.

iii. Computation of the nearest neighbors using Euclidean distance measure. Features used in the computation were taken from ARFF file (Figure 3.4); “src_byte” is the initial byte sent to the destination host from the source; while “dst_byte” constitutes the bytes sent from destination host to the source. The computations demonstrate the reason why ensemble learning by Random Forest was used to update the misclassified points in the base

learning models, especially as the training was done sequentially, starting with K-NN and the corrected sample_weights form a new set of data, which is used to train DT in the next iteration.Test objects will be assigned following plurality voting system by the base models. Where there is a tie, the predictions are sorted and the first one taken as the accepted prediction. Mix values of different features in different propotions of k were also used to compute the nearest neighbors. In it, the target object is assigned the class with the majority vote.

All the attributes in @ATTRIBUTE statement of ARFF file are open to both normal applications and malware, and each record in the @DATA subsection is a defined data type (normal or any of the attack types).

3.5.1.1 Computation of Compromised Systems During an Attack Process

To demonstrate the use of mini-max normalization, significant features from ARFF file are selected and used to compute compromised systems during an attack. Only attack data types are used in the computation. The attack types include smurf (DOS) attack, ipsweep (probe attack), warezclient (R2L) attack, and buffer_overflow (U2R) using min_max normalization whose formula appears in Equation 3.1

$$U_i = \frac{V_i - X_1}{X_2 - X_1} (Y_2 - Y_1) + Y_1 \qquad \text{Equation 3.1}$$

Where

X_1, X_2: are the minimum and maximum boundaries of a feature

Y_1, Y_2: are the new scale at which normalization is done, for example a range of [0,1]

V_i: the value of the attribute as indicated in the @DATA section of ARFF file.

U_i: the computed value of the normalized attribute to fall within the range.

To apply this model on the NSL-KDD dataset, features from the ARFF file were used from the four attack types as contained in Table 3.6.

These parameters are applied in Equation 3.1 where

X_1: is the src_byte feature, and is the initial number of bytes sent to the destination host.

X_2: is the count feature, which is the number of times the destination host is connected

Y_1: is the initial logged_in feature, which is usually zero (0)

Y_2: is the successfully logged in status, which is one (1)

V_i: is the destination host count with the same IP address

U_i: is the number of compromised systems in a connection session of not more than two Seconds.

Table 3.6: Features used to compute compromised systems (Min-Max normalization)

Src_byte	Count	Dst_host_count	Logged_in	
X_1	X_2	V_i	Y_1	Y_2
smurf (DOS Attack)				
520	184	255	0	1
ipsweep, Probe attack type				
8	1	2	0	1
warezclient, R2L attack type				
334	1	5	0	1
buffer_overflow, U2R attack				
220	1	53	0	1

Using smurf, a DOS attack type, with the following extracted values from Figure 3.5, as represented in Table 3.6

X_1: = 520

X_2 = 184

Y_1: = 0

Y_2 = 1,

$V_i = 255$,

U_i: is the number of compromised systems being computed.

Therefore,

$$Ui = \frac{255 - 520}{184 - 520}(1 - 0) + 0 = \frac{-265}{-336} = \mathbf{0.79}$$

That is, the number of systems compromised and added to the botnet family, within the two seconds DOS attack window is 0.79, which is within the expected range [0, 1].

Using ipsweep, a probe attack type, the number of compromised systems computed using the following features from Figure 3.4:

X_1: = 8;

$X_2 = 1$,

Y_1: = 0,

$Y_2 = 1$,

$V_i = 2.0$,

$$Ui = \frac{2 - 8}{1 - 8}(1 - 0) + 0 = \frac{-6}{-7} = 0.86$$

This also is within the normalized range, 0.86.

To compute the number of local systems compromised by a remote attacker (R2L), warezclient attack type, the following features from Figure 3.4 file were used, as depicted in Table 3.6.

X_1: = 334;

$X_2 = 1$,

Y_1: = 0,

$Y_2 = 1$,

$V_i = 5$,

$$Ui = \frac{5 - 334}{1 - 334}(1 - 0) + 0 = \frac{-329}{-333} = 0.99$$

The compromised system in this case is 0.99, and falls within the range [0, 1].

Using the privileged attacker, buffer overflow (internal attacker), which is user to root (U2R) attack type; the number of systems that could be compromised within the two seconds window were computed. The parameters used were also extracted from Figure 3.5 (ARFF file), as depicted in Table 3.6.

X1 = 220

X2 = 1

Y1 = 0

Y2 = 1

Vi = 53

$$Ui = \frac{53 - 220}{1 - 220}(1-0) + 0 = \frac{-167}{-219} = \mathbf{0.76}$$

Out of the four attack methods, the rate of compromise by an internal attacker is lower than the other three; warezclient (R2L) attack type had the maximum hit of 0.99.

3.5.1.2 Computation of Principal Components (PCs) Using PCA

Similarly, to reduce the level of ambiguity and other redundant features such as outliers, noise, etc. computation of principal component using normal and ipsweep (probe attack type) features was demonstrated using PCA in two-dimensional linear process. PCA reduces dimension of the features by computing Eigen vectors with large Eigen values, given a threshold (>= 70%) to eliminate redundant features, outliers or noise, and select acceptable principal components. An *m x n* covariance matrix **A,** with elements being the features of the training dataset, an identity matrix **I** and a column vector **V** were used in the computation. From the characteristic polynomial, the determinant of the said polynomial is derived, called characteristic equation, as shown in Equation 3.2.

$$AV = \lambda IV$$

$$(A - \lambda I)V = 0$$

$$\text{Det}\,(A - \lambda I) = 0 \qquad \text{Equation 3.2}$$

Where

A: is the m x n covariance matrix

V: is the number of rows or objects (a column vector)

I: is the identity matrix

λ: is a scalar called Eigen value

m x n: m being the observations (rows) and n is the features (columns)

Det (A - λI) = 0: indicates that the matrix used is a singular matrix. The elements used for matrix **A** are normal type and ipsweep (probe attack type). The features used from the ARFF file for the above data types were "count" and "srv_count".

Top row: (normal data)

2: "count" (integer value) (column 23)

2: "srv_count" (integer value) (column 24)

Second row (ipsweep: probe attack)

1: "count" int. (column 23)

1: "srv_count" int. (column 24)

These values are extracted from the @DATA subsection of the ARFF file. They represent normal data and ipsweep attack type in Figure 3.6

0,tcp,ftp_data,SF,43,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,255,62,0.24,0.02,0.01,0,0.87,0.97,0,0,normal,14 *

1,tcp,ftp,RSTO,0,133,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,1,1,0,0,2,2,0.5,1,0.5,1,0,0,1,1,ipsweep,9 *

Figure 3.6: ARFF Records From Where Count and Srv_count Values were Randomly Extracted (Columns 23 and 24 Respectively)

$$\mathbf{A} = \begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix}$$

$$\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ \lambda I = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}$$

$$\text{Det } \mathbf{(A - \lambda I) = 0}$$

$$\text{Det}\left(\begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}\right) = 0$$

$$\text{Det}\begin{bmatrix} 2-\lambda & 2 \\ 1 & 1-\lambda \end{bmatrix} = 0$$

$$(2 - \lambda)(1 - \lambda) - 2 \text{ x } 1$$

$$(2 - \lambda)(1 - \lambda) - 2 = 0$$

$$2 - 2\lambda - \lambda + \lambda^2 - 2 = 0$$

$$= \lambda^2 - 3\lambda = 0$$

$$= \lambda(\lambda - 3) = 0$$

$$\lambda = 0$$

$$\lambda - 3 = 0$$

$$\lambda = 3$$

Compute the associated Eigen vectors.

Let the associated Eigen vector be $\mathbf{V} = \begin{bmatrix} Y \\ Z \end{bmatrix}$

Therefore, for $\lambda = 0$

$$(A - \lambda I)\, V = 0$$

$$\begin{bmatrix} 2 - \lambda & 2 \\ 1 & 1 - \lambda \end{bmatrix} \begin{bmatrix} Y \\ Z \end{bmatrix} = 0$$

$$\begin{bmatrix} 2 - 0 & 2 \\ 1 & 1 - 0 \end{bmatrix} \begin{bmatrix} Y \\ Z \end{bmatrix} = 0$$

$$\begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} Y \\ Z \end{bmatrix} = 0$$

$$2Y + 2Z = 0$$

$$Y + Z = 0$$

Let Y = 1

Then $1 + Z = 0$

$$Z = -1$$

Hence, $V_0 = \begin{bmatrix} +1 \\ -1 \end{bmatrix}$

For $\lambda = 3$

$$\begin{bmatrix} 2 - 3 & 2 \\ 1 & 1 - 3 \end{bmatrix} \begin{bmatrix} Y \\ Z \end{bmatrix} = 0$$

$$\begin{bmatrix} -1 & 2 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} Y \\ Z \end{bmatrix} = 0$$

$$-Y + 2Z = 0$$

Let Y = 1

$$-1 + 2Z = 0$$

$$2Z = 1$$

$$Z = ½$$

$$V_3 = \begin{bmatrix} 1 \\ ½ \end{bmatrix}$$

The threshold is set to >= 70%, and the spectrum of A is σ (**A**) = (0+3) = 3

$$\frac{3}{3} x \frac{100}{1} = 100\%$$

Therefore, $\mathbf{V_3}$ with Eigen value 3 produced the first principal component (PC).

The pre-processed data of NSL-KDD dataset contained in ARFF file (Figure 3.4) will be analyzed using production rules and Inference Engine to determine how data types were derived; and how IRS uses production rules (contained in knowledge base) to determine the counter measure to use during an attack.

Principal Component Analysis (PCA) Algorithm

PCA algorithm reduces the dimension of data features by computing the principal components. The steps used include:

Input parameter:

A: this is Covariance matrix

I: this is identity matrix

ⲗ : Lambda, a constant called Eigen value

V: this is the Eigen vector

i. From the characteristic polynomial, form the characteristic equation., which is the determinant as illustrated in Equation 3.3

$$Det (A - ⲗI) = 0 \qquad \text{Equation 3.3}$$

ii. Expand the determinant using row factor form

iii. Solve the resulting quadratic equation to get the Eigen values of the covariant matrix A

iv. Find the corresponding Eigen vectors V, to the Eigen values

v. Sort the Eigen vectors in ascending order, according to their corresponding Eigen values

vi. Set a threshold to use, say ≥ 70%, of Eigen values

vii. Select Eigen Vectors with the highest Eigen values within the threshold range

viii. The selected Eigen Vectors represent the principal components (PC). The first PC is assigned the Eigen Vector with the highest Eigen value, in that order.

3.5.1.3 Classification of Target Object Using K- Nearest Neighbor (KNN)

The distance measure used in this research work is the Euclidean distance measure; and optimal parameter k = 5 points represents the nearest neighbors with the shortest distance to the target point. The target object is assigned the class with majority votes among the nearest neighbors. The model is illustrated in Equation 3.4

$$D_{Euc(l}(p, q) = \sqrt{\Sigma_{i=1}^{n} (p_i - q_i)^2} \qquad \text{Equation 3.4}$$

Where p and q are points in n-space.

The value of k used in this work is k = 5.

The computations that follow illustrate the possibility of all values of k being from the same class. To illustrate the computation of nearest neighbors, Euclidean distance measure is used, which determines the nearest neighbors of the training dataset; data used in this illustration are gotten from ARFF file, Figure 3.4. Src_byte is the initial byte sent from the source to the destination host in the current layer, and dst_byte is the bytes sent to the source from the destination host. The Euclidean distance measure is an extension of Pythagoras theorem thus:

$$D_{Eucl}(xi, yi) = \sqrt{(x1 - y1)^2 + (x2 - y2)^2 + , ..., + (xn - yn)^2}$$

$$= \sqrt{\Sigma_{i=1}^{n} (xi - yi)^2}$$

Where

X_i: is the initial source byte (src_byte) sent to the destination host

Y_i: is the bytes sent from the destination host (dst_byte)

D (xi, yi): is the distance measure using Euclidean distance formula.

Table 3.7 depicts the selected values of src_byte and dst_byte, from columns 5 and six of normal records (as contained in the @ATTRIBUTE subsection) of ARFF.

Table 3.7: Attribute Extracted From Normal Data Records of ARFF File

Src_byte	Dst_byte
34	0
5	381
19	0
22	75
30	0

Computing nearest neighbors of the same class using Euclidean distance measure, with k = 5; that is, five different points in the cluster (all of them being normal data type). From Table 3.7, the following bytes were extracted from normal (benign) records and used in computing the nearest distance:

$$D_{Eucl}(x_5,y_5) = \sqrt{(34-0)^2+(5-381)^2+(19-0)^2+(22-75)^2+(30-0)^2}$$

$$= \sqrt{(34)^2+(-376)^2+(19)^2+(-53)^2+(30)^2}$$

$$\sqrt{1156+141{,}376+361+2{,}809+900}$$

$$= \sqrt{146{,}602}$$

$$D(x_5, y_5) = 382.89 \text{ bytes}$$

Where

X_5: is the src_byte from source to destination host

Y_5: is the dst_byte from the destination host to source

D_{Eucl} (x_5, y_5): is the computed result using Euclidean distance measure.

Using guesspassword (R2L) attack type in five different points (k = 5), with the same features: Src_byte and dst_byte values from the records. From Table 3.8 the following values are extracted randomly from ARFF file in Figure 3.4

Table 3.8: Attributes Extracted From Guesspassword Attack Type of ARFF File

Src_byte	Dst_byte
26	93
28	103
32	93
30	93
123	178

$D_{Eucl}(x_5,y_5)=$

$$\sqrt{(26-93)^2+(28-103)^2+(32-93)^2+(30-93)^2+(123-178)^2}$$

$$=\sqrt{(-67)^2+(-75)^2+(-61)^2+(-63)^2+(-55)^}$$

$$=\sqrt{4{,}489+5{,}625+5{,}721+3{,}969+3{,}025}$$

$$=\sqrt{22{,}829}$$

D (x_5,y_5) = 151.10 bytes.

In a similar way, nearest neighbors were computed using src-byte and dst-byte values from mscan (probe) attack type records as depicted in Table 3.9.

Table 3.9: Attributes Extracted From Mscan Attack Type of ARFF File

Src_byte	Dst_byte
24	715
0	44
0	36
0	15

6 54

$$D_{Eucl}(x_5, y)_5 =$$

$$\sqrt{(24-715)^2+(0-44)^2+(0-36)^2+(0-15)^2+(6-54)^2}$$

$$=$$

$$\sqrt{(-691)^2+(-44)^2+(-36)^2+(-15)^2+(-48)^2}$$

$$=\sqrt{477{,}481+1{,}936+1{,}296+225+2{,}304}$$

$$=\sqrt{483{,}242}$$

$D_{Eucl}(x_5, y_5) = 695.16$ bytes

From literature, the class with the shortest distance forms the nearest neighbors. In which case, guess_password, which is a remote to local attack type, is the nearest neighbor with a smaller value of 151.10 bytes, against normal features with distance value of 382.29 bytes, and mscan (probe), with a value of 695.16 bytes. The drawback of this illustration, when members of the nearest neighbors are all of the same class, is that the test target will be assigned to the class, as there are no other features of the opposite class to compare. KNN is biased to the majority class.

Although rare, the painted scenario is possible and would not give the desired result if KNN is faced with one class nearest neighbors; bearing in mind that we are dealing with a binary problem.

Situations where K=5 with the ratio of 3:2 and 4:1 are also considered.

Compute the nearest neighbors using Euclidean distance measure with K=5, using normal data type with 3 points and snmpgetattack, a R2L attack type with 2 points in the cluster of nearest neighbors. The data features extracted are depicted in Table 3.10.

Table 3.10: Attributes Extracted from Normal Data Records and Snmpgetattack Type of ARFF File; Ratio of 3:2

Src_byte	Dst_byte
Normal class labels	
46	86
45	82
52	54
Snmpgetattck class label	
105	146
105	0

$$D_{Eucl}(x_5,y_5) = \sqrt{(46-86)^2+(45-82)^2+(52-54)^2+(105-146)^2+(105-0)^2}$$

$$=\sqrt{(-40)^2+(-37)^2+(-2)^2+(-41)^2+(105)^2}$$

$$=\sqrt{1{,}600+1{,}369+4+1{,}681+11{,}025}$$

$$=\sqrt{15{,}679}$$

$$D(x_5, y_5) = 125.22 \text{ bytes}$$

Where

X_5: is the src-byte of normal data type with 3 points, and snmpgetattack (R2L) with 2 points respectively.

Y_5: is the dst-byte of normal data type with 3 points, and snmpgetattack (R2L) data type with 2 points respectively.

$D_{Eucl}(x_5, y_5)$: is the distance measure computed using Euclidean measure

Now we compute the nearest neighbors using normal data type with 3 points and processtable (DOS) attack type with 2 points in the cluster of K=5; and src_byte and dst_byte as features. Table 3.11 illustrates the random records extracted from Figure 3.4.

Table 3.11: Attributes extracted from Normal data records and Processtable of ARFF file; in a ratio of 3:2

Src_byte	Dst_byte
Normal class labels	
46	77
35	91
43	0
Processtable class labels	
0	15
0	44

$$D_{Eucl} = (x_3, y_2) = \sqrt{(46 - 77)^2 + (35 - 91)^2 + (43 - 0)^2 + (0 - 15)^2 + (0 - 44)^2}$$

$$= \sqrt{(-31)^2 + (-56)^2 + (43)^2 + (-15)^2 + (-44)^2}$$

$$= \sqrt{961 + 3136 + 1849 + 225 + 1936}$$

$$= \sqrt{8{,}107}$$

$$D\ (x_3, y_2) = 90.04 \text{ bytes}$$

Where

X_5 : is the src-byte of normal data type with 3 points, and processtable (DOS) attack type with 2 points respectively.

Y_5: is the dst-byte of normal data type with 3 points and processtable (DOS) data type with 2 points respectively.

$D_{Eucl}\ (x_5, y_5)$: is the computed distance in bytes

The next example combines attack types: smurf (DOS) having 3 points in the cluster and satan (probe) data types using 2 points in the cluster, with features used being src_byte and dst_byte. Table 3.12 depicts the extractions from the labels in Figure 3.5.

Table 3.12: Attributes extracted from smurf and satan data records of ARFF file, in a ratio of 3:2

Src_byte	Dst_byte
Smurf	
508	0
520	0
1008	0
Satan class labels	
20	0
9	196

$$D_{Eucl}(x_5, y_5) =$$

$$\sqrt{(508-0)^2+(520-0)^2+(1008-0)^2+(20-0)^2+(9-196)^2}$$

$$=\sqrt{(508)^2+(520)^2+(1008)^2+(20)^2+(-187)^2}$$

$$=\sqrt{258{,}064+270{,}400+1{,}016{,}064+400+34{,}969}$$

$$=\sqrt{1{,}579{,}897}$$

$$D_{Eucl}(x_5, y_5) = 1{,}256.94 \text{ bytes}$$

Where

X_5: is src_byte of smurf (DOS) attack data type with 3 points, and Satan (probe) attack type with 2 points respectively

Y_5: is dst_byte of Smurf (DOS) attack type with 3 points, and Satan (probe) attack data type with 2 points in the cluster respectively

$D_{Eucl}(x_5, y_5)$: is the computed result using Euclidean distance measure

From the computations, it is observed that Normal/processtable relationship has the shortest distance of 90.04 bytes. From this set, the Normal type has the majority vote of 3. The target object is then assigned to Normal.

The combination with the shortest distance forms the nearest neighbors. In this case, the combination of normal data type and processtable (DOS) data type constitutes the nearest neighbors, with a distance (D_{Eucl} (x_5, y_5) being 90.04 bytes. The test object will be assigned normal class since it forms the majority vote. The src_byte (x-values) and dst_byte (y-values) are randomly chosen from the data subsection of ARFF file of NSL-KDD dataset. The classification of the target object is depicted in Figure 3.7

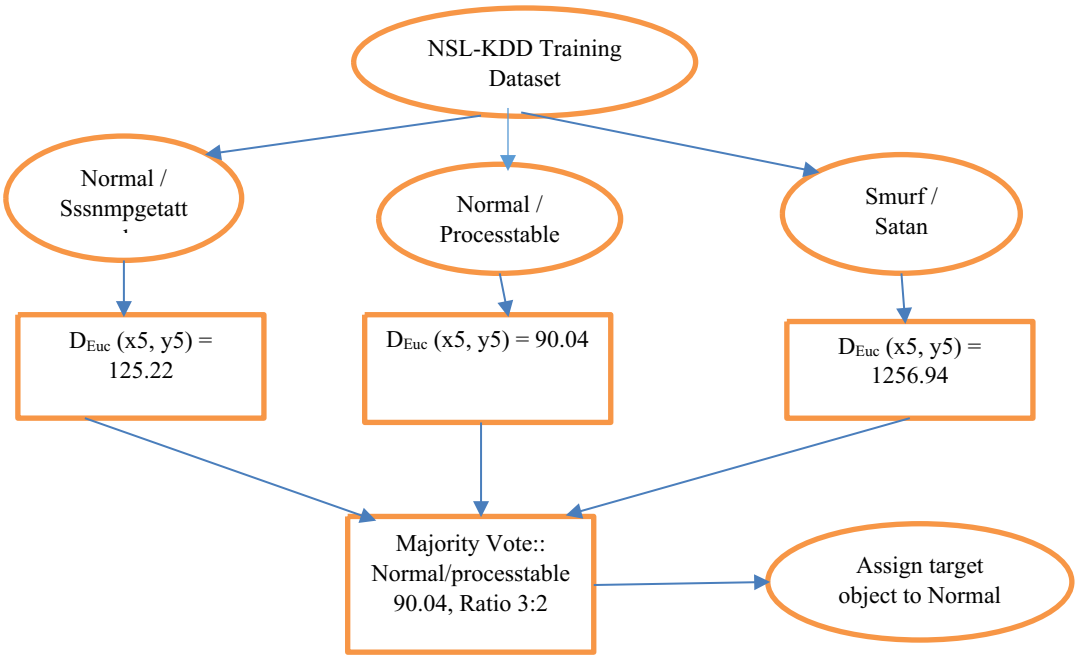


Figure 3.7: Analysis and Classification of Target Object Using KNN

Algorithm of KNN

The main steps used to classify the target object in KNN include:

i. Determine the number of nearest neighbors (the k value)

ii. Compute the distance between the target point and all training samples using Euclidean distance measure

iii. Sort the distances and determine the nearest neighbors based on the k^{th} minimum distance

iv. Assemble the classes of the nearest neighbors

v. Assign the target (test) object to the class with majority votes.

3.5.2 Data Preprocessing

One-Hot Encoder transforms Categorical Variables

To process the categorical features in the dataset, encoding was employed to transform these features into numerical representations, thereby creating additional features. Specifically, Label Encoding was applied to convert the 'protocol_type', 'service', and 'flag' columns into numerical values. Following this, One-Hot Encoding was utilized to further encode these categorical features, generating a binary column for each category within 'protocol_type', 'service', and 'flag'. Figure 3.8 depicts the dataset before encoding.

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	...
0	5	tcp	smtp	SF	2429	475	0	0	0	0	...
1	0	udp	domain_u	SF	45	134	0	0	0	0	...
2	0	udp	domain_u	SF	45	80	0	0	0	0	...
3	1979	udp	other	SF	145	105	0	0	0	0	...
4	14462	tcp	other	RSTR	1	0	0	0	0	0	...

Figure 3.8: Categorical variables before encoding

The resultant encoded features were then concatenated with the original dataset as a new feature set, which increased the dimension of the dataset to 126 columns, and the

	protocol_type_tcp	protocol_type_udp	flag_REJ	flag_RSTO	flag_RSTOS0	flag_RSTR	flag_S0	flag_S1	flag_S2	flag_S3	flag_SF	flag_SH
0	1	0	0	0	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	0	0	1	0
2	1	0	0	0	0	0	1	0	0	0	0	0
3	1	0	0	0	0	0	0	0	0	0	1	0
4	1	0	0	0	0	0	0	0	0	0	1	0

initial categorical columns were subsequently removed to finalize the feature engineering process. Figure 3.9 depicts the dataset after encoding process.

Figure 3.9: The dataset after one-hot-encoding

On the other hand, the target variables “attack types” were manually encoded such that each attack group are mapped to a numerical value {0: ‘DOS’, 1: ‘PROBE’, 2: ‘R2L’, 3: ‘U2R’, and 4: ‘NORMAL’}.

Balancing the Dataset

The unbalanced class dataset underwent further processing to achieve balance, which is crucial to avoid bias in model performance. Using the Synthetic Minority Over-sampling Technique (SMOTE) from the `imbalance` library, the minority classes were oversampled to match the number of instances in the majority class. Initially, the class distribution showed a significant imbalance, with the 'normal' and 'DOS' classes having a much higher count compared to 'PROBE', 'R2L', and 'U2R'. After applying SMOTE, the dataset was balanced, resulting in an equal number of instances (67,342) for each class. This balanced dataset is crucial for training a model that performs well across all classes. Figure 3.10 provides a visual representation of the class distribution before and after the balancing process.

UNBALANCED ATTACKS GROUPS	BALANCED ATTACKS GROUPS

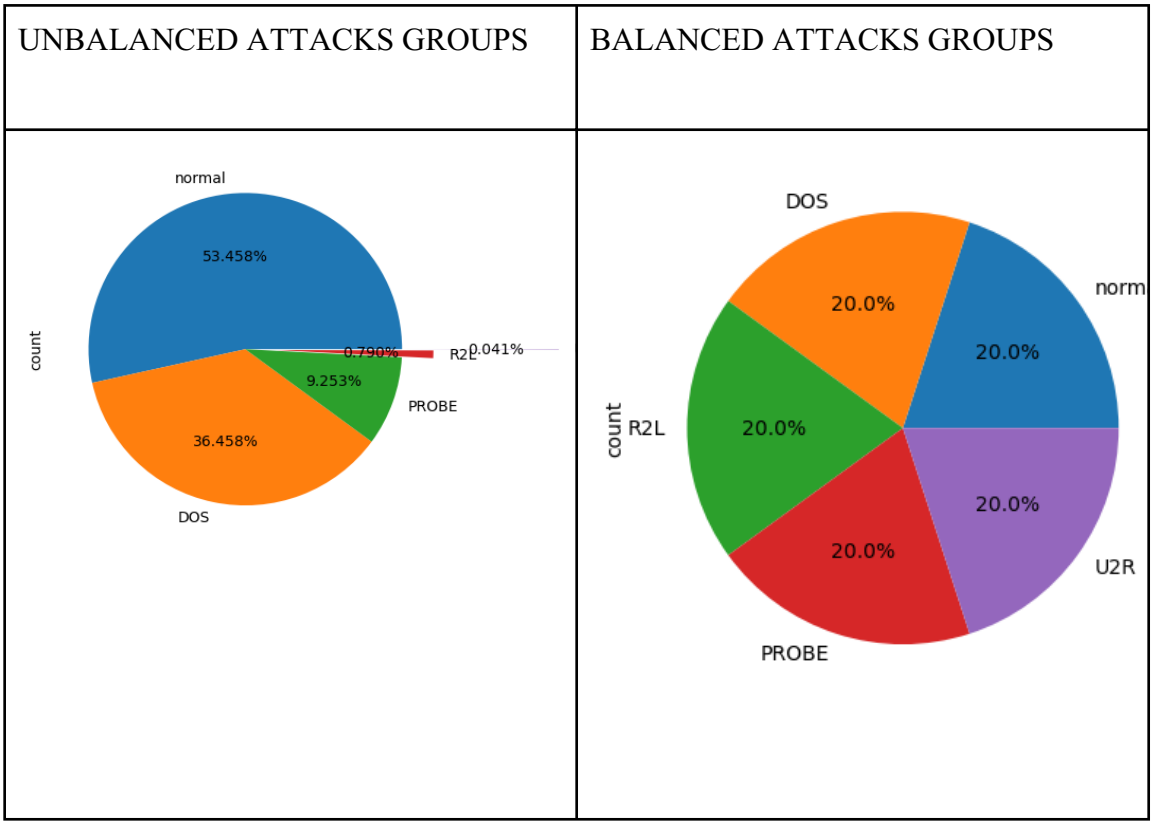


Figure 3.10: Pie Charts Showing Unbalanced and Balanced Attack Types
Source: Researcher (2024)

Principal Component Analysis (PCA)

Criteria for selecting the number of components

Given the high-dimensional dataset, dimensionality reduction was applied using Principal Component Analysis (PCA). Initially, the data was standardized to ensure each feature contributed equally to the analysis. The standardized data was then transformed using PCA to identify the principal components. The explained variance ratio was analyzed to determine the number of components required to capture at least 95% of the total variance (95% being the threshold or cut off point). A plot of the cumulative explained variance against the number of principal components was generated to visualize this relationship. Based on this plot, the number of components that explained at least 95% of the variance was selected. This dimensionality reduction step is essential to enhance model efficiency and performance by reducing the complexity of the dataset while retaining most of the information.

Figure 3.11 illustrates the explained variance against the number of principal components, highlighting the point where 95% of the variance is captured.

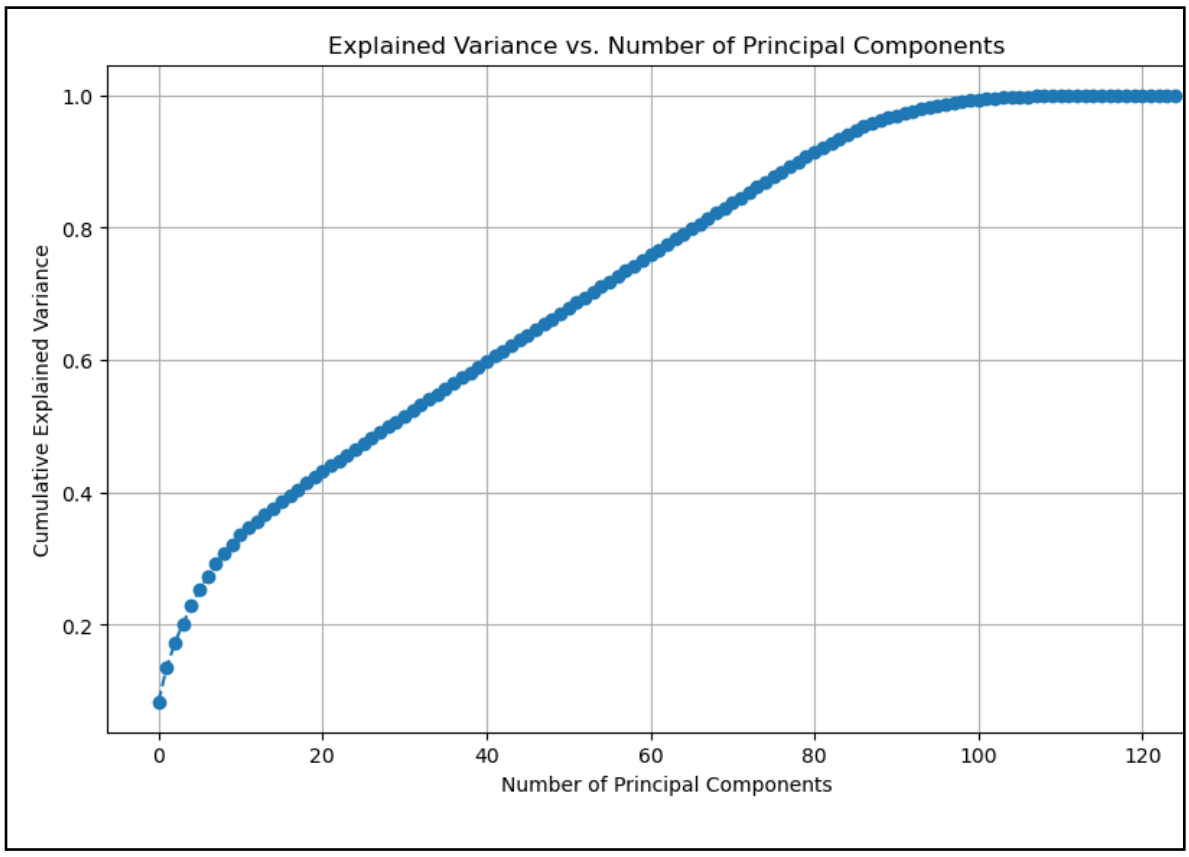


Figure 3.11: Explained variance against the principal components
Source: Researcher (2024)

Data Segmentation

After processing the dataset through sampling and balancing using SMOTE, the number of data points was increased to 336,710, ensuring equal representation for each class. Categorical encoding expanded the columns from 42 to 125, and PCA was applied to reduce the dimensions to 88 columns, retaining 95% of the variance. Due to computational constraints, a subset of 100,000 balanced data points was selected. The dataset was then split into 80% for training and 20% for testing, ensuring the model was trained on a large portion of the data to learn patterns effectively, while the testing set

provided independent validation to assess the model's accuracy and generalization capabilities. This approach ensures robust evaluation and effective predictions on new data.

3.5.3 Model Training

K-Nearest Neighbor (KNN) Classifier

Following the data segmentation strategy, 80% of the dataset was used for training, while the remaining 20% was reserved for testing. The KNN classifier was initialized with 5 neighbors and utilized the Minkowski distance metric, equivalent to the Euclidean distance measure with $p = 2$. After training the model, predictions were made on the validation dataset. These predictions were evaluated using a custom `evaluate_model` function, with performance visualized through a confusion matrix and a detailed classification report, highlighting metrics such as precision, recall, and F1-score.

Random Forest Classifier (RF)

To train and evaluate a Random Forest model, it was initialized with 500 estimators to ensure robust and stable model performance by aggregating the predictions from multiple decision trees, thus reducing overfitting. Entropy was used to measure the quality of splits, focusing on maximizing the information gain at each split. The model was trained on the training dataset (80%), and predictions were made on the test dataset (20%). To assess the model's performance, evaluate_model() function was used, and printed the classification report, and visualized the results using a confusion matrix

Naive Bayes Classifier (GNB)

To training and evaluating a Naive Bayes model, Gaussian Naive Bayes classifier was used due to its effectiveness in handling continuous data and its assumption of a normal

distribution for the features. The model was trained on the training dataset (80%), and predictions were made on the test dataset (20%).

Decision Tree Classifier (J45)

To train a Decision Tree model, entropy was used as the criterion to measure the quality of splits, focusing on maximizing the information gain at each split. The model was trained on the training dataset (80%), and predictions were made on the test dataset (20%).

Logistic Regression (LR) Classifier

To train Logistic Regression model, a maximum of 1000 points were initialized and iterated to ensure convergence. The model was trained on the training dataset, which comprised 80% of the total data, and predictions were made on the test dataset, which comprised the remaining 20%.

Ensemble Learning (Voting Classifier)

The ensemble learning approach employed in this implementation leverages the strengths of multiple individual classifiers to enhance predictive performance. The voting classifier is the core of this strategy, combining five distinct models: K-Nearest Neighbors (KNN), Random Forest (RF), Naive Bayes (GNB), Decision Tree (J45) and Logistic Regression (LR). Soft voting is utilized, where the predicted class probabilities from each model are averaged, and the class with the highest average probability is selected. This ensemble method ensures a more balanced and robust prediction. The weights assigned to each model—slightly favoring KNN and Decision Tree—reflect their relative contributions to the ensemble, fine-tuning the overall performance. The process involves defining the ensemble model with these estimators, training it on the dataset, predicting outcomes on the validation set, and evaluating its performance using key metrics such as accuracy, precision, recall, F1-Score, and ROC-AUC

3.5.4 Algorithm Used to Develop the Framework

To develop the framework that will be used to detect the menace of malware threats, the following are considered:

i. Obtain the data to train the models, which is NSL-KDD dataset, from Kaggle.

ii. Balance the data types (labels) using SMOTE technique

iii. Pre-process the balanced dataset using mini-max normalization, principal component analysis (PCA), One_hot_encoder(), Label_encoder().

iv. Split the dataset into training dataset (80%) and test dataset (20%) using train_test_split() function, in python programming language

v. Train base classifiers such as KNN, NB, DT, and LR in parallel using Random ForestClassifier() function

vi. Predict the labels using each trained classifier with model.predict() function

vii. Aggregate the predictions of the trained models using votingClassifier() function in sklearn.ensemble module to form a consensus classifier called soft vote.

viii. Train the soft vote classifier and use it to predict test dataset, and produce results.

   a. Compare the predictions of soft vote with the actual labels using confusion matrix

   b. Compute the standard metrics such as accuracy, precision, recall and f1-score, and area under the curve (AUC).

ix. Send reported alerts in the form of Normal and attack types such as DOS, Probe, R2L and U2R to the user via the Hospital.

x. Display results on the desktop or laptop or smartphone used.

3.6 Expert System

In artificial intelligence, an expert system is a computer system emulating the decision-making ability of a human expert. Expert systems are designed to solve complex problems by reasoning through bodies of knowledge, represented mainly as if–then rules rather than through conventional procedural programming code. Expert systems have four main components:

i. Data base which consists of ARFF dataset

ii. Knowledge base (Production rules)

iii. Inference Engine (represented by AIRS)

iv. User Interface

Figure 3.12 depicts an expert system.

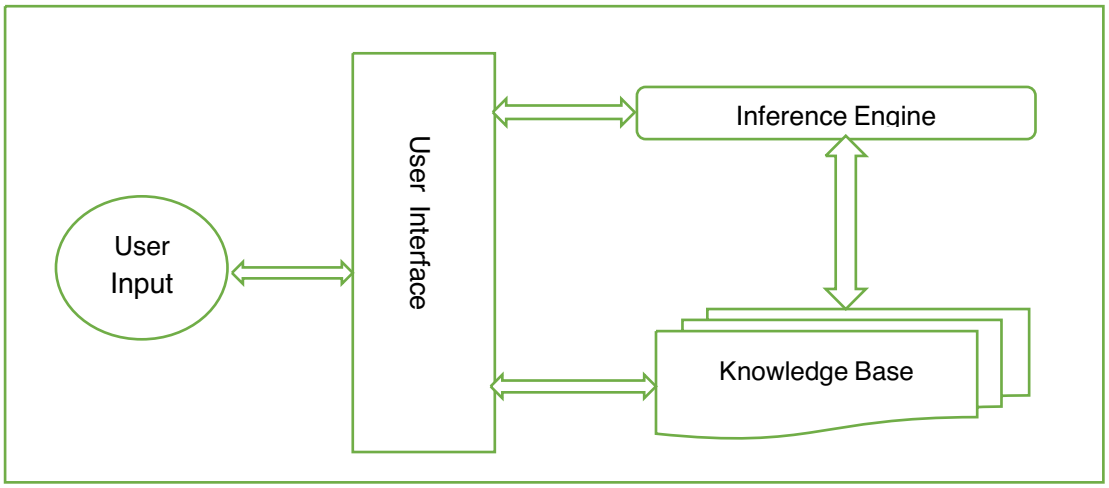


Figure 3.12: An expert system illustrating four basic components
Source: Researcher (2024)

3.6.1 Knowledge Base (Production Rules)

The knowledge base comprises of production rules represented in the form of IF..THEN statements. The features that determine if an application is normal or malware are symbolic features such as protocols, services and flags. They are represented in the

knowledge base using production rules. The protocols are tcp, udp and icmp; there are many service types such as aol, auth, bgp, and so on; so also are flags which include OTH, REJ, RSTO, S0, S1, S2, S3.

The rules, which are IF …THEN statements, logically relate information in the IF part to that of the THEN part using:

IF {Condition} THEN {Application} rules.

Some of the production rules are depicted in Table 3.13 comprising thirty-four rules.

Table 3:13: Production Rules Used to Determine the Class of Applications

S/N	Production Rules
I	IF protocol = tcp AND service = http AND flag = SF THN Application = normal; OR
Ii	IF protocol = udp AND service = domain_u AND flag = SF THEN App = normal, OR
Iii	IF protocol = tcp AND service = pop_3 AND flag = SF THEN App = guess- password (R2L) OR
Iv	IF protocol = tcp AND service = telnet AND flag = RSTO THEN App = guesspassword (R2L),
V	IF protocol = tcp AND service = ftp AND flag = RSTO THEN App = ipsweep (Probe attack type)
Vi	IF protocol = udp AND service = other AND flag = SF THEN App = snmpgetattack (R2L)
Vii	IF protocol = tcp AND service = telnet AND flag = S3 THEN App = processtable (DOS attack type)
Viii	IF protocol = tcp AND service = smtp AND flag = SF THEN App = mailbomb (R2L attack type)
Ix	IF protocol = tcp AND service = image4 AND flag = RSTO THEN App = mscan (Probe attack type) OR
X	IF protocol = tcp AND service = pop_3 AND flag = RSTO THEN App = mscan (Probe attack type) OR
Xi	xi. IF protocol = tcp AND service = telnet AND flag = RSTO THEN App = mscan (Probe attack type).
	% Policy rules for Neptune (DOS attack type)
Xii	IF protocol = tcp AND service = private AND flag = REJ THEN App = Neptune (DOS attack type) OR
Xiii	IF protocol = tcp AND service = private AND flag = S0 THEN App = Neptune OR
Xiv	IF protocol = tcp AND service = remote_job AND flag = S0 THEN App = Neptune (DOS attack type) OR

Xv	IF protocol = tcp AND service = name AND flag = S0 THEN App = Neptune OR
Xvi	IF protocol = tcp AND service = netbios_ns AND flag = S0 THEN App = Neptune OR
Xvii	IF protocol = tcp AND service = mtp AND flag = S0 THEN App = Neptune OR
Xviii	IF protocol = tcp AND service = finger AND flag = S0 THEN App = Neptune OR
Xix	IF protocol = tcp AND service = supdup AND flag = S0 THEN App = Neptune OR
Xx	IF protocol = tcp AND service = uucp_path AND flag = S0 THEN App = Neptune OR
Xxi	IF protocol = tcp AND service = z39_50 AND flag = S0 THEN App = Neptune OR
Xxii	IF protocol = tcp AND service = csnet_ns AND flag = S0 THEN App = Neptune OR
Xxiii	IF protocol = tcp AND service = efs AND flag = S0 THEN App = Neptune (DOS)
	% Policy rules for data types sharing the same symbolic features
Xxiv	IF protocol = udp AND service = private AND flag = SF THEN App IN {normal, teardrop, snmpguess, snmpgetattack, satan}
Xxv	IF protocol = tcp AND service = ftp_data AND flag = SF THEN App IN {normal, warezclient, warezmaster}
Xxvi	IF protocol = tcp AND service = finger AND flag = SF THEN App IN {normal, Satan}
Xxvii	IF protocol = tcp AND service = telnet AND flag = SF THEN App IN {processtable, guess_password, mscan}
Xxviii	IF protocol = udp AND service = other AND flag = SF THEN App IN {Satan, snmpgetattack, multihop}
Xxix	IF protocol = tcp AND service = ftp AND flag = SF THEN App IN {warezmaster, buffer_overflow, guesspassword}
Xxx	IF protocol = icmp AND service = ceo_i AND flag = SF THEN App IN {Satan, ipsweep, saint}
Xxxi	IF protocol = icmp AND service = ecr_i AND flag = SF THEN App IN {smurf, pod}
	% Policy rules for AIRS (Automated Intrusion Response System) % ON {Event} IF {Condition} THEN {Action} % Aggressive action policy
xxxii	ON ANOMALY DETECTION IF role != IN {SA, DBA,CEO} and Source IP 192.168.0.0/16 and

	Object type = Table and Object name IN {Server, Host, Network} and Command IN {READ_SMS, WRITE_SMS, SEND_SMS, MODIFY_PHONE_STATE} Hardware Apps IN {GPS, GPRS, CAMERA, CONTACT LIST} THEN Abort and PRINT {Action = Abort}
	% Policy rule for conservative action, where the target is not so important to the Organization
Xxxiii	ON ANOMALY DETECTION IF Confidence metric LOW and Target Object NOT CRITICAL THEN NOP and PRINT {Action = ignore}
	% This policy re-authenticates the privileged user (inside attacker)
Xxxiv	ON ANOMALY DETECTION IF request IN {Contact list, media filter, log file, GPS, internet filter} and INTENT IN {connectivity_change, uninstall_shortcut, view_phone_state} THEN put on_hold and CONFIRM {second authentication} ON SUCCESS NOP and PRINT {Action = Continue} ON FAILURE Abort and Revoke PRINT {Action = Abort, Revoke}

These rules are traversed when the inference engine receives facts from the user through the user interface or anomalous report from IDS.

3.6.2 Inference Engine (IE)

Inference engine is a device used by Expert system to reason by matching facts supplied by the user, through the user interface, with the rules formulated and stored in the knowledge base and then draws conclusion.

It has two reasoning processes:

i. Backward chaining (modus tolens), which starts reasoning from the solution or goal, and works backward to find facts that support the goal; tests some rules that are relevant to the problem at hand; it is bottom-up

and goal-driven. It is suitable for problems that start from a hypothesis, such as medical diagnosis of a patient.

ii. Forward chaining (modus ponens) starts from the set of facts and moves towards the conclusion or solution.

In this research work, forward chaining will be used to process the facts that match with the rules. It will be used because it is suitable for problems that start with data collection such as planning, monitoring and control; it starts with initial facts, tests all the rules, and it is data-driven. Figure 3.14 illustrates the sequence of events in forward chaining process as modified from Al-Ajlan (2015).

From Figure 3.13 the sequence is mimicked as follows:

Referring to rule number **xxxiii** in the knowledge base**,** the anomalous application is requesting for dangerous permissions and IE scans the rules and found that the request is dangerous, and aborts the application. “Abort” will be reported to the user.

Secondly, assuming IE receives this information from the ARFF file,

protocol = tcp, service = telnet and flag = RSTO

it will scan the rule base beginning from the first rule, which will not match, it checks the next rule, which will still not match; and then the next rule will not match. It is rule number **iv** that matches and guesspassword (R2L) will be added to the working memory (WM). Once the rule has fired, then IE goes back to the WM for another input from ARFF file until there are no more inputs from the file. If all the 33 rules do not match the set of variables in a particular record, IE will then return to the user, through the user interface, and ask for information about the set of values presented.

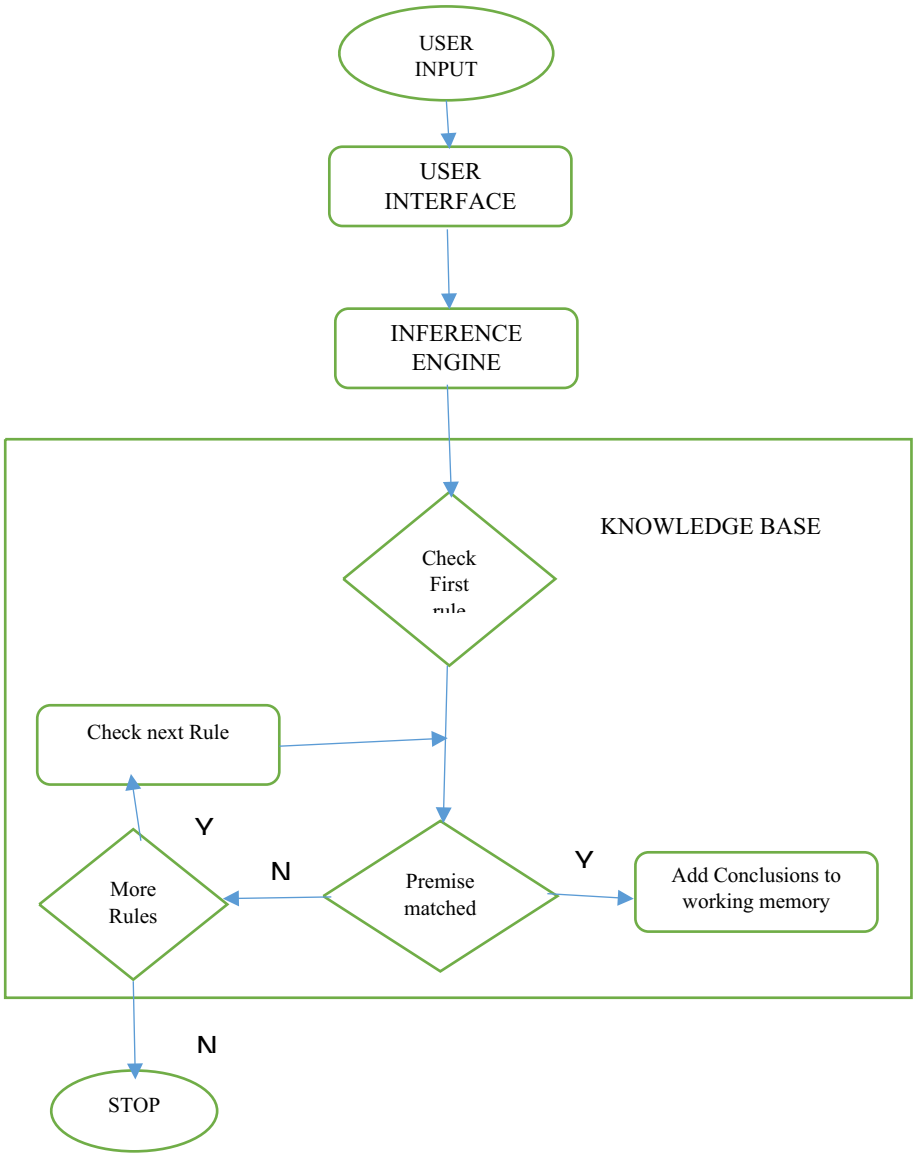


Figure 3. 13: Forward chaining inference process.
Source: Researcher (2024)

3.6.3 The User Interface

The user interface enables the communication between the user and the Inference engine, to perform specific tasks. It is also required to translate and interpret the input for the application of rules. It is depicted in Figure 3.14. From Figure 3.14, the problem to be solved is stated at the top of the user interface; the user first requests to use the

system by selecting the system from the menu. The system then requests for user details, and the user enters his or her details through the user interface, which will be validated by the system. If the user's details are not correct, he or she will be given only two more chances to input the correct details or access to the interface will be denied. The user in turn enters the facts of the problem in the space below for use by the inference engine to match with the rules in the knowledge base. The results of the system are communicated to the user through the *record label* and *action taken by IE* boxes.

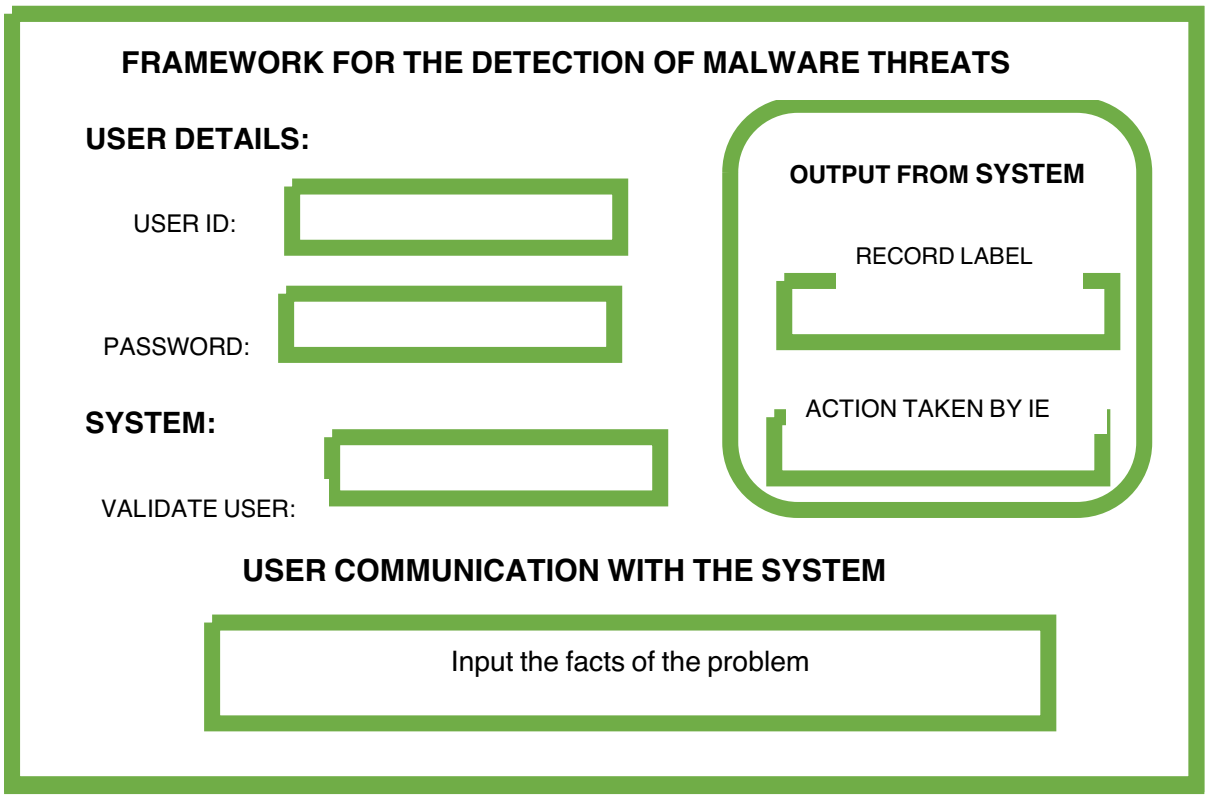


Figure 3.14: The User Interface for Communication Between User and Inference Engine

3.7 System Design

The objective of this research work is to design a Framework that will monitor applications seeking to use the EHR system to either install third party applications or to mischievously disrupt the operations and steal vital information after gaining access to the system. Malware developers usually request for more permissions to be

mischievous when the need arises, especially making use of dangerous permissions. For the Framework to be up and doing, Ensemble learning, piloted by Random Forest, is used to train base (weak) models in parallel, and use a voting technique to ascertain the cooperation of the separate models that predicted labels independently.

The system design is implemented using unified modeling language (UML). In this research work, four system design components such as user interface (UI), use case diagram, class diagram and sequence diagram will be designed using UML. They include:

(i) Use case diagram (dynamic)

(ii) Class diagram (static)

(iii) Sequence diagram (dynamic)

(iv) User interface

These components interact with the system through the user interface.

3.7.1 Use Case Diagram

A use case diagram specifies the interactions of external objects (Actors), which are either human or applications or systems, with the system being developed. Actors used in this work are depicted in Table 3.14.

Table 3.14: Actors Definition

S/N	ACTOR NAME	DESCRIPTION
1.	User	This actor uses the laptop to link with the system by logging in and provide user details, through the user interface. The user may decide to cancel the process.
2.	Administrator	This actor is a privileged user (the hospital) that links the user with the system, after

		verifying the user's details. Admin also gives the user privilege to update his or her details and provides the enabling environment for logging out.
3.	The CSP Server	This actor is a repository where updates of the user details are stored and, confirmation of the user details are made when required.

Assumptions

i. It is assumed that the user has subscribed with at least one network operator to get access to the internet

ii. The laptop used in this work is assumed to use Windows operating system

iii. Given that the analysis is dynamic using CCT, data collected are uploaded to the cloud for processing

Figure 3.15 depicts use case diagram.

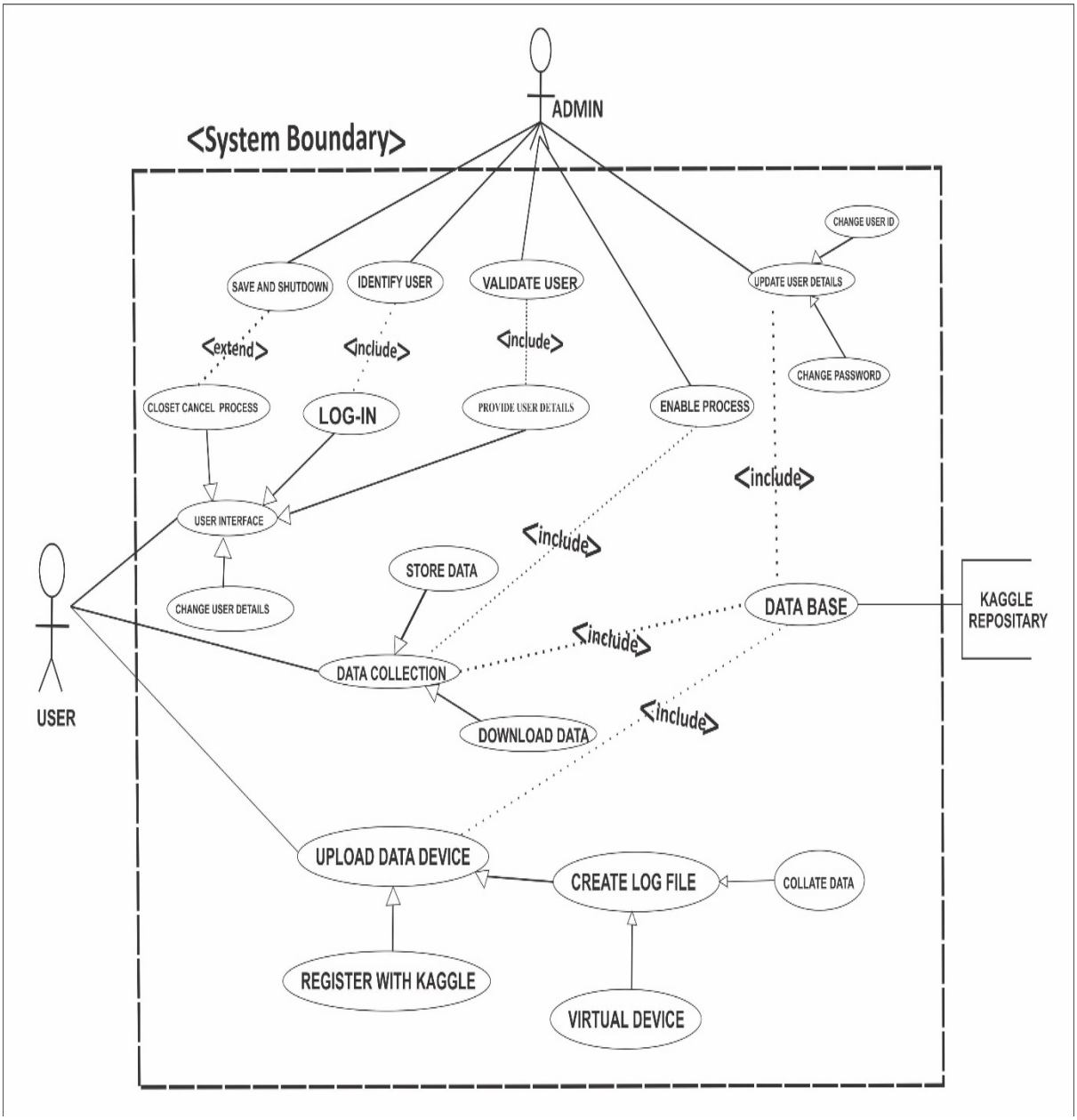


Figure 3.15: Use Case Diagram
Source: Researcher (2024)

Table 3.15, Table 3.16, and Table 3.17 depict use case definitions

Table 3.15: Use Case Definition 1

USE CASE NAME	CODE	DEFINITIONS
User Interface	UC001	The user accesses the system through the user interface by providing identity details as he or she logs_in.
Actors		User, Administrator and CSP Server repository

Events in Use Case Definition 1

EVENT	ACTOR	SYSTEM
	1) This use case is initiated when the user navigates to the E-Hospital platform.	2) The system requests for identification.
	3) The user logs in to the platform by providing the following details (i) user ID (ii) Password	4) The system validates the user. 5) Displays the E-Hospital platform with menus. 6) From the menu, the customer has the option of Editing the user ID or password or cancel the process or activate any process like loading data to the system memory.
	7) However, the user can decide to cancel the transactions.	8) The system terminates the process and signs out the user, then shuts down.

Table 3.16: Use Case Definition 2

USE CASE NAME	USE CASE CODE	DESCRIPTION
Data collection	UC002	The user collects the data to use and train/test models from CSP server repository and stores in the device.
Actors involved.	User, Admin, CSP Server repository.	

Events in Use Case Definition 2

Customer	System
Step 1: From the menu, the user chooses to download HER from CSP Server repository	2). The system enables the process.
	3) CSP effects the transfer of the dataset through the internet
Step 4: The user then stores the received dataset in the device memory	6) The system signs out the user
5) The user concludes the process.	

Table 3.17: Use Case Definition 3

USE CASE NAME	CODE	DESCRIPTION
Upload data	UC003	Given the constraints of the device, the user then registers with CSP using the address: https://www.kaggle.com in order to access kaggle's console and use it to process transactions in the cloud
Actors		User, Admin, and CSP Server repository.

Events in Use Case Definition 3

ACTOR	SYSTEM
Step 1: This use case starts with the user registering with Kaggle online.	2) The system provides enabling environment
	3) Kaggle asks for user identity.
step 4: The user provides details such as user ID and password.	
	5) Kaggle registers the user and provide him or her access to its console
Step 6:	
The user opens a log file and collates both data and virtual image of the device.	7) The system provides the enabling environment
Step 8) The user uploads the log file to the proxy server	
	10) The proxy server in turn forwards the log file to the cloud for processing.
Step 9) The user saves a copy of the log file in the device	

3.7.2 Class Diagram

A class diagram provides the structural (static) view of the system.

Classes are represented by boxes, which are subdivided into three parts:

- The class name
- The attributes
- The method (behavior)

In most cases, the attribute subsection contains private features, which are manipulated by the methods of the class only; while the methods are mostly public or protected, where they can be called by other classes or sub classes.

The visibility elements mentioned above are summarized below

- Private - used by the class methods only

+ Public - can be called by other classes' methods

Protected – called by the class and sub classes only

Mobility values indicate the number of objects each class uses or participate in a session. They are illustrated in Table 3.18.

Table 3.18: Multiplicity Symbols and Meanings

Symbols	Meaning
0	None
1	One
M	An integer value
0..1	Zero or one
0..*	Zero or more
1..*	One or more
*	Any non-negative integer (Zero or more)

Figure 3.16 depicts the class diagram of user interface process

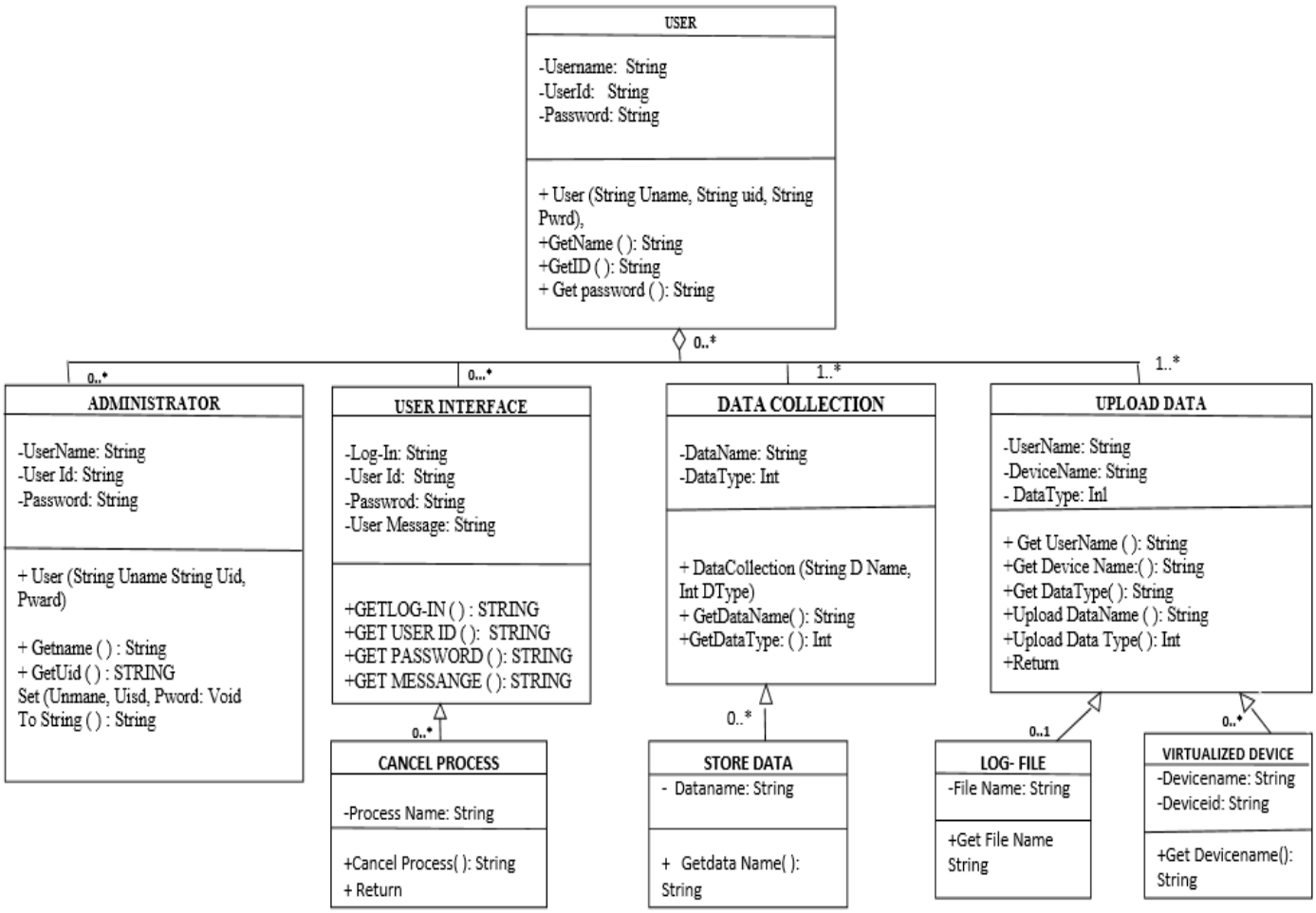


Figure 3.16: Class Diagram
Source: Researcher (2024)

Table 3.19 depicts the process relationships in the Class diagram

Table 3.19: Process Relationship in the Class Diagram

CLASS	CLAUSES
User	A single user can aggregate 0 or more Transactions.
Administrator	This is a surrogate to the system, but as well a user and can access one or more processes.
User Interface	This is the gateway between the user and the system. It accesses one user at a time. It has a child subclass, cancel process, which inherits the features of the parent.
Data Collection	Data collection is related with the system and database, therefore one or more processes can be accessed. It has a child subclass, store process, which inherits its features and methods (that are either public or protected) but can over-write any method of the parent.
Upload Data	This process is aggregated to the user and relates with the admin and database where data is downloaded from and updated. It has two child processes, log-file and virtualize device
Cancel Process	This child class is a subclass to user interface and can inherit its features and methods, but has a right to over-write any method. It accesses zero or more processes.
Store Data	This subclass is a child class of Data collection use case, and can inherit all the methods and features of the parent class. It can access zero or more objects.
Log-file	This process is a sub class to upload Data process, and can inherit its features and methods. It has zero to one mobility.
Virtualize Device	This process is a sub class to upload Data process, and can inherit its features and methods. It has zero to one mobility.

3.7.3 Sequence Diagram

A sequence diagram models the interactions between objects in a single use case. In this research work, the sequence diagram of user interface process is illustrated in

Figure 3.17. It illustrates how the user, system admin and CSP Server repository interact with each other to carry out a function.

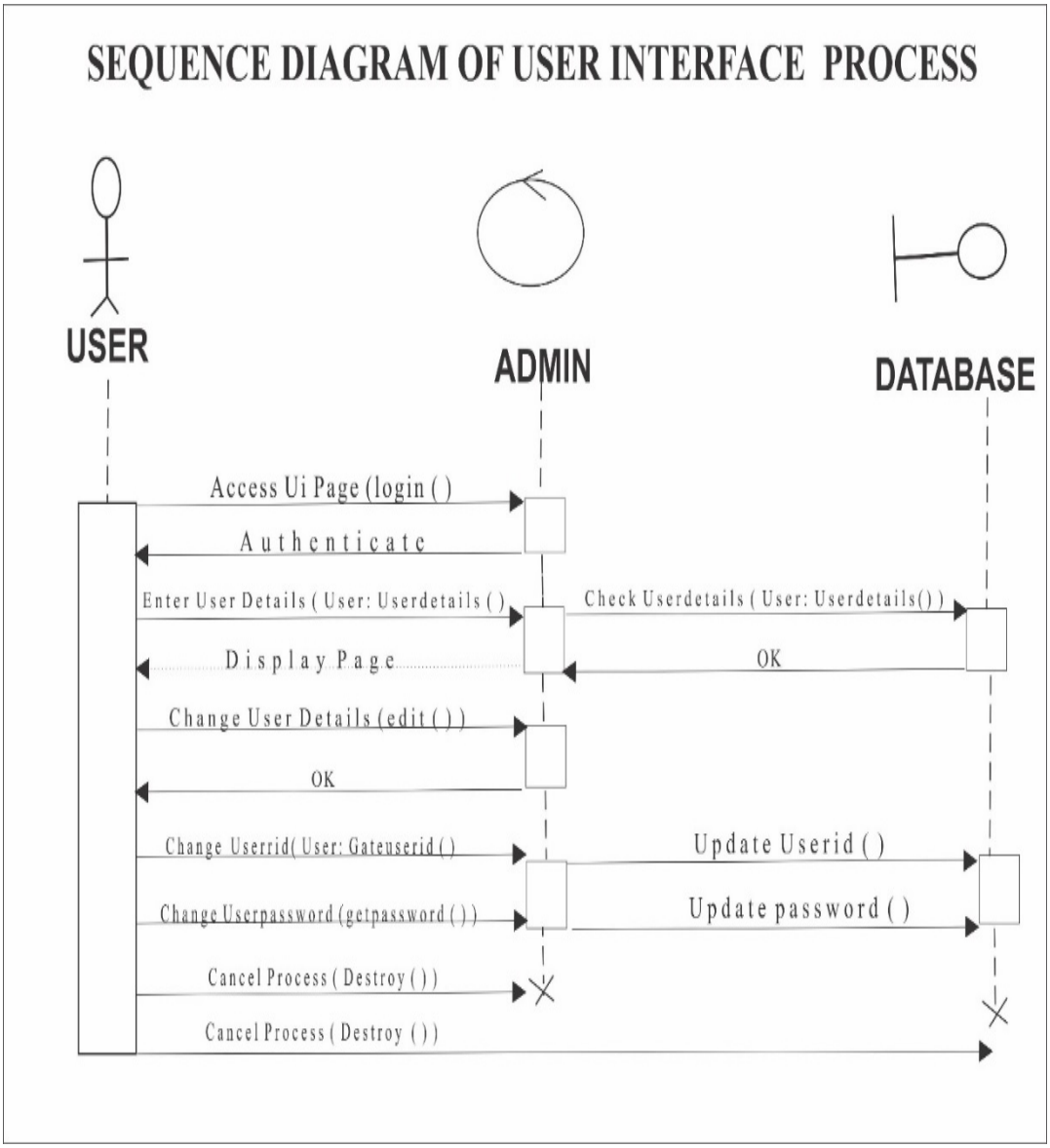


Figure 3. 17: Sequence Diagram
Source: Researcher (2024).

Sequence Diagram Details on User Interface Process

The following are the steps or interactions between the user and the system

i. The user initiates access to the user interface (UI)

ii. The system requests for user identity

iii. The user provides user details:

- UserID ( ): string
- Password ( ): string

iv. The system verifies user details from the database

checkUserDetail ( )

v. User decides to change his or her details:

getUserID ( ): string

getPassword ( ): string

vi. Administrator updates the database

updateDatabase ( )

vii. User decides to cancel the process:

destroy ( )

viii. The system updates the database:

updateDatabase ( )

CHAPTER FOUR
RESULTS AND DISCUSSION

4.1 Preamble

In this Chapter, the results, analysis of the results using confusion matrix and discussion of the results are presented. This research work leverages on the security of the healthcare system and patient's privacy in the cloud, and designs and implements a framework to analyze and detect unfriendly applications that would undermine security and abuse the privacy of patients' records, using machine learning tools. Given the sophistication of malware threats, research community approves the combination of classifiers to produce a robust classifier to detect the stealthy malware threats and safeguard the electronic health records of patients. Ensemble learning technique, bagging, using random forest as the training algorithm is adopted in this research work. The base classifiers include KNN, DT, NB and LR. The predictions of these models are aggregated using votingclassifier() to produce a consensus classifier (soft vote model), which is trained and used to analyze the test dataset and produce results.

The models are further evaluated using cross validation techniques to assess their performances using the following cross validation parameters: accuracy, precision, recall, f1-score and area under the curve (AUC). The soft vote Classifier demonstrates exceptional performance, with metrics indicating near-perfect predictive capabilities. The model achieves an accuracy of 0.99788, precision of 0.997879, recall of 0.997885, F1-Score of 0.997880, and a ROC-AUC score of 0.99996. The confusion matrix further supports these results, showing minimal misclassifications and indicating that the majority of instances are correctly classified, using TP, FP, TN and FN rates. The AUC-ROC scores for each class include—normal (1.0), DOS (0.99995), PROBE (0.99995),

R2L (1.0), and U2R (0.99991). The generated values of these parameters form part of this chapter. They will be used to compare with the results of similar works in literature.

4.2 System evaluation

The primary objective of this thesis is to leverage machine learning models to efficiently detect and classify network attacks. The system architecture begins with the NSL-KDD dataset, followed by data extraction and initial analysis. Manual experimentation is conducted to explore features, and data preparation includes label encoding, normalization, and PCA for dimensionality reduction. Five machine learning models: K-Nearest Neighbors (KNN), Random Forest Classifier (RF), Decision Tree (DT) Classifier, Naive Bayes (NB) Classifier and Logistic regression (LR) are trained on the characteristics of normal applications, and to use that knowledge to classify malware threat applications from normal ones. Each model was evaluated based on its ability to identify patterns and anomalies in network traffic. The ultimate goal was to create an ensemble model using these classifiers with soft voting, allowing each model to contribute to the final decision by averaging their predicted probabilities. More predictive power was assigned to the best-performing models within the ensemble to enhance overall accuracy and efficiency. The predicted outputs (results) were compared with the expected labels using confusion matrix, with metrics such as TP, TN, FP and FN, being a supervised learning problem.

The models were assessed using cross validation metrics such as accuracy, precision, recall, F1-Score, and area under the (receiver operating characteristics) curve (AUC-ROC), with particular attention to AUC-ROC scores for each attack class. Additionally, Principal Component Analysis (PCA) was applied for dimensionality reduction, and SMOTE was used to balance the dataset, ensuring robust and unbiased model performance. This comprehensive approach aims to develop a highly accurate and

reliable machine learning model for detecting and classifying network attacks, thereby enhancing cyber security measures, and protecting electronic health records system for improved morbidity and mortality rates of healthcare system. The framework will assist decision makers in planning and funding electronic healthcare system, which has relative advantages of electronic health records (EHR) over the traditional paper system still mostly used in Nigerian health systems.

Some advantages of EHS over the paper-based system include

i. Easy collaboration and data sharing within and between hospital staff
ii. Mobility
iii. Cost reduction on ICT services
iv. "Pay for what you use", rather than owning expensive infrastructure and equipment
v. Scalability
vi. Business continuity
vii. Flexibility
viii. Strategic values.

4.3 Results

The results are derived from classified test dataset consisting of actual and predicted data classes such as: NORMAL, DOS, PROBE, U2R and R2L. As a supervised learning problem, the actual values of this test dataset are known (Github Inc., 2020). Subsequently, PCA was applied with the chosen number of components, and the resulting principal components were used to create a new DataFrame with records and columns as depicted in Table 4.1. Given the new dataFrame, the original features have been replaced with the derived principal component PC1 TO PC90 including the labels.

The predicted labels of the soft vote model will be matched with the expected (or actual or known) class types for analysis by confusion matrix.

The extract presented in Table 4.1 has only twenty records out of 125,973 records, for want of space. However, some misclassifications are found such as: in lines 290 Normal label was misclassified as U2R; in line 3530, Normal label was misclassified as PROBE; in line 3770, Normal label was misclassified as DOS. Similarly, in line 4132, Normal label was misclassified as DOS respectively. Observe that extracts from the results sheet are presented in sets of eight columns and ten rows each with labels as "class" and "predicted" as depicted in the last group of columns in the first ten (10) rows or records in Table 4.1. The last group of columns in each set contains the class or actual and predicted labels. This is a convenient way of presenting the ninety columns, of the set of records extracted, including label column.

Table 4.1: Extracts of Results Presented in Groups of Records and Columns

SN	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
5188	-1.43341	0.596405	-0.44467	-1.14996	2.509284	1.120042	-1.56106	0.849499
110012	1.208092	0.588724	0.065598	-2.68224	3.190255	1.539098	-2.00343	1.376987
289664	4.76409	-4.17956	-0.4102	0.516971	0.221604	0.315511	-0.40474	0.2378
11733	-3.80554	0.220176	-6.16195	0.148521	-1.61079	0.749816	-0.13229	-0.60043
112761	4.891938	-4.20786	-0.46565	0.473168	0.239561	0.282929	-0.35332	0.18615
91777	-1.45234	-0.159	1.378551	-0.01186	0.491553	-0.4992	0.621893	-0.73715
241358	-2.60999	0.093091	0.456637	1.193181	0.238186	-1.14077	0.08406	1.828881
18724	-1.99626	-0.54995	-6.02778	-4.54614	0.817659	0.947084	1.188159	-4.17533
37521	-0.59901	0.710094	0.353775	-0.04459	-0.22459	0.124985	0.222975	-0.80542

PC9	PC10	PC11	PC12	PC13	PC14	PC15	PC16
0.064919	-0.36444	0.981542	-0.09704	0.566431	-1.09105	-0.30802	-1.8841
0.27099	-1.39076	0.896866	-0.73486	0.056956	0.346718	0.048024	0.775836
0.55475	-0.44988	-0.40682	-0.66058	0.008154	0.413063	0.305829	0.972755
0.506227	-1.29716	0.044736	0.301274	-0.06884	-0.01851	-0.0492	-0.14783
0.370871	-0.41412	-0.39389	-0.65759	0.039806	0.403971	0.2998	0.940998
0.197757	-1.81841	-1.07062	-2.35901	1.654002	-0.73425	-0.37695	-1.69693
-0.35664	1.213269	0.734716	-0.20698	0.142829	-0.01658	0.121971	0.250291
-3.16717	8.379447	-5.16169	-2.3807	0.301356	0.105053	0.057338	0.442796
0.485391	-2.60376	-0.95341	-2.17076	1.730463	-0.94782	-0.43489	-1.98836

PC17	PC18	PC19	PC20	PC21	PC22	PC23	PC24
-0.10106	-0.21132	0.521873	-1.12022	0.028723	-0.14176	-0.23805	-0.04097
0.451745	-0.62111	-0.14419	-0.1341	0.058582	0.01413	0.108368	-0.06636
0.277142	-0.23485	-0.17009	-0.07654	0.10105	-0.00301	-0.04474	-0.08857
-0.21819	-0.02033	0.109183	0.029106	0.036225	0.056989	0.027328	-0.08436
0.250505	-0.23763	-0.15985	-0.08607	0.11368	0.012623	-0.04428	-0.12272
-0.06101	-0.68544	-0.83662	4.697652	-1.31443	-0.97666	-0.29589	0.162373
-0.09655	0.033966	-0.03281	0.147765	-0.03313	0.142869	0.183885	-0.06653
0.67874	0.099866	-0.19613	0.368079	-0.18241	-0.07104	-0.03548	-0.16956
-0.10621	-0.77626	-0.86778	5.51174	-1.51989	-1.06084	-0.31204	0.106846

PC25	PC26	PC27	PC28	PC29	PC30	PC31	PC32
-0.24601	0.274055	-0.02524	-0.31038	-0.5044	0.464046	0.248291	0.274289
0.006556	0.029092	0.01416	-0.03842	-0.03363	0.093504	-0.02029	0.013292
-0.10364	0.024803	-0.00698	-0.03687	-0.07626	0.071422	0.110993	0.108795
-0.03596	0.087156	0.024951	-0.01998	-0.07275	0.032027	0.064439	0.056664
-0.13227	0.02922	-0.01486	-0.03268	-0.0861	0.095472	0.118817	0.132343
0.637566	0.206886	-0.00399	0.119879	0.321275	-0.32297	0.428508	-0.64849
0.171206	0.108496	0.065695	-0.08492	-0.01409	0.028414	-0.06793	-0.19687
-0.07961	0.190489	-0.01736	-0.32991	-0.43887	0.237063	0.279467	0.299154
0.716157	0.290157	-0.0125	0.08961	0.370457	-0.38355	0.43724	-0.65779

PC33	PC34	PC35	PC36	PC37	PC38	PC39	PC40
0.168795	0.196434	0.180756	0.124714	-0.03128	0.276966	-0.21958	0.164281
-0.03607	-0.02418	-0.01627	-0.01714	0.014987	0.003005	0.017857	0.001184
-0.02033	0.052084	0.02941	-0.0094	0.009499	0.022714	-0.03871	-0.01296
-0.03152	0.056972	0.015423	0.008694	0.02103	0.046384	-0.00665	0.026593
-0.02636	0.055578	0.021791	-0.00643	0.008927	0.028292	-0.04374	-0.00978
-0.09869	-0.27961	-0.02805	0.31	0.020309	-0.23659	0.029365	-0.26532
-0.09783	-0.16139	-0.04988	-0.05709	0.059904	-0.14054	0.056844	-0.05366
-0.02813	0.127981	0.308976	0.054738	-0.03959	0.247038	-0.0705	0.01809
-0.14986	-0.31848	-0.0607	0.353178	0.040466	-0.20971	0.032624	-0.28506

PC41	PC42	PC43	PC44	PC45	PC46	PC47	PC48
0.271315	0.298299	-0.21565	0.161457	-0.11932	0.092712	0.019399	0.01545
0.01087	0.000783	0.083221	-0.00975	-0.01198	0.015618	0.004404	0.00741
-0.0519	0.014324	0.176439	0.007044	-0.01006	-0.0017	-0.0012	0.006924
0.072205	0.069764	0.052549	-0.01969	0.025628	0.005628	-0.00473	0.003232
-0.03571	0.001406	0.18288	-0.00516	-0.01508	0.000759	-0.00132	0.007851
0.170807	-0.025	0.047128	0.05009	-0.11704	-0.08584	0.006675	0.007316
0.010414	-0.0402	0.038305	-0.02805	-0.02097	0.036438	0.009034	-0.00609
0.126591	0.353941	-0.34637	0.132045	0.196084	-0.2348	-0.13636	-0.00628
0.232716	0.003749	0.022029	0.034817	-0.11401	-0.09436	0.010222	0.001772

PC49	PC50	PC51	PC52	PC53	PC54	PC55	PC56
0.013883	-0.01929	-0.09379	0.033215	0.048525	-0.0909	-0.05882	-0.10556
0.002983	-0.0018	0.006215	0.014866	-0.01186	-0.01966	-0.01573	-0.01319
-0.00822	-0.00056	-0.01273	-0.00421	0.021861	-0.00565	0.002936	0.046428
0.00501	-0.00183	-0.00391	0.025313	-0.01366	-0.01784	0.004542	-0.00928
-0.00818	-0.00074	-0.0133	0.005196	0.025573	-0.00801	0.004914	0.039688
-0.05436	-0.0034	-0.00966	-0.0073	-0.24796	0.175255	-0.0329	-0.18207
0.001456	0.00142	0.034947	-0.00052	-0.02172	-0.0286	-0.00334	-0.02392
0.013532	0.032552	-0.03622	-0.14356	-0.03896	-0.00622	-0.03715	0.006501
-0.04338	-0.00754	-0.00837	0.043197	-0.31472	0.191177	-0.02769	-0.20914

PC57	PC58	PC59	PC60	PC61	PC62	PC63	PC64
-0.20741	-0.00829	-0.19157	-0.03852	-0.02104	-0.05698	-0.06674	0.026465
0.002367	-0.01418	0.000238	-0.0067	-0.00202	-0.0009	0.010973	0.003148
0.049624	-0.01655	0.013147	-0.00464	-0.01317	-0.03436	0.027717	0.037506
-0.00951	-0.01931	-0.02858	-0.01658	0.002502	0.014562	0.004828	0.017342
0.045607	-0.02449	0.011386	-0.00897	-0.00885	-0.02353	0.03172	0.038085
0.008792	-0.00169	0.052061	0.03718	-0.05403	0.061843	0.022992	-0.02295
0.020733	0.003774	0.004646	0.008505	0.010498	0.0128	0.020946	-0.00428
-0.00617	0.034188	-0.00812	-0.00269	-0.03655	-0.13021	-0.02872	0.019974
-0.00162	0.003402	0.026731	0.060524	-0.05699	0.105273	0.012538	-0.0362

PC65	PC66	PC67	PC68	PC69	PC70	PC71	PC72
-0.03984	0.131631	-0.431	-0.09089	0.145141	-0.04534	-0.02046	-0.05586
0.039914	0.005963	0.04663	0.017481	-0.04266	0.025823	-0.00114	0.009557
0.086355	-0.02088	0.115364	0.046035	-0.08443	0.030532	0.0008	0.000616
0.070286	0.009329	-0.01818	0.005931	0.017778	0.007867	-0.01791	-0.00064
0.093789	-0.0201	0.118175	0.050006	-0.08391	0.034083	0.001436	0.002349
0.134086	-0.19176	0.102926	0.074559	-0.05577	-0.00405	0.007944	0.012066
0.022551	-0.01414	0.021962	0.00581	0.013413	0.00299	-0.01391	0.005417
-0.003	-0.0029	-0.01204	-0.0191	-0.06942	-0.00806	-0.01469	-0.00198
0.114101	-0.18693	0.076515	0.073868	-0.01479	-0.01284	0.00777	0.007949

PC73	PC74	PC75	PC76	PC77	PC78	PC79	PC80
-0.00547	0.232443	-0.01008	0.022562	-0.32387	-0.23442	1.0158	0.187921
-0.00674	-0.05749	0.000146	0.004887	0.102742	0.022052	-0.25433	-0.10798
-0.00571	-0.11614	-0.00223	0.01973	0.178398	0.029836	-0.51407	-0.15209
0.016878	0.00158	0.000748	-0.01906	-0.02431	0.004828	-0.01447	-0.0347
-0.00636	-0.11437	0.000229	0.015693	0.172527	0.034322	-0.52383	-0.15292
0.071227	-0.00677	-0.00635	-0.05522	-0.09755	0.019701	-0.12645	0.03126
0.013893	-0.00031	-0.00818	-0.02002	-0.03588	0.014368	0.013182	0.000854
-0.01928	-0.05049	-0.00889	0.037035	0.107675	-0.02587	-0.10588	-0.05275
0.099113	0.030275	-0.01287	-0.07317	-0.18672	0.016234	0.001142	0.066488

PC81	PC82	PC83	PC84	PC85	PC86	PC87	PC88	class	Predicted_
0.042053	-0.24301	-0.06745	0.331163	-0.51075	0.21299	-0.04361	1.33562	normal	normal
-0.05209	-0.6451	0.110284	0.252581	-0.59601	0.269111	0.228987	1.851491	DOS	DOS
-0.05794	-0.81644	0.267012	0.230452	-1.01476	0.258604	0.561143	-0.09836	U2R	U2R
-0.0068	-0.04942	-0.01854	0.010824	-0.13111	0.027481	-0.0334	-0.12842	DOS	DOS
-0.06084	-0.81933	0.265786	0.247251	-1.06038	0.265955	0.583791	-0.2435	normal	normal
-0.01655	-0.05213	-0.04446	-0.04408	0.168358	-0.02896	0.016122	-0.00182	normal	normal
-0.00846	-0.05918	0.0466	0.036568	-0.05721	-0.00211	0.010228	0.008848	R2L	R2L
-0.03255	-0.58336	0.127527	0.312746	-0.14551	0.064929	0.105298	0.175536	DOS	DOS
0.004521	0.308484	-0.03548	-0.19775	0.293882	-0.10385	-0.08376	0.216704	PROBE	PROBE

Rows 11 to 20 begin here.

93402	-2.00847	-0.32392	1.344167	-0.87271	0.447848	1.599749	6.407752	0.151383
313978	-0.38632	0.051189	-0.43239	-3.18599	3.636681	1.018746	-1.24182	0.588119
103367	-1.24601	-0.03116	2.347593	1.742339	1.105891	1.548678	-1.07418	-1.5806
165945	4.917583	-4.20257	-0.43597	0.513373	0.215858	0.327084	-0.41618	0.242556
1011	-3.76339	0.337634	-6.36657	0.88681	-1.86574	1.122255	-0.72474	-0.09987
124722	-2.23267	0.229103	0.52391	0.947295	0.169123	-1.04238	0.053781	1.841169
323856	-2.49873	0.16529	0.395036	1.059523	0.269006	-1.2546	0.221194	1.788027
80180	5.283799	3.648818	-0.69371	-0.54698	-1.47147	-1.30429	1.479508	-0.43894
213785	4.961381	-4.19194	-0.42442	0.542991	0.203266	0.31619	-0.42019	0.275789
58917	-2.39428	0.447401	-4.31487	0.243806	-1.26502	1.160017	-0.92368	0.214541

0.35398	-1.47465	-1.28346	0.110311	-0.13034	-1.33129	-3.19277	0.916946
0.638574	-0.3355	0.333779	-0.69904	-0.02105	0.520495	0.258204	1.033567
-0.78028	-0.34204	-1.09888	0.490023	0.244657	-0.01601	0.099692	0.035288
0.408203	-0.43496	-0.37666	-0.6552	0.043991	0.397146	0.298004	0.943952
0.369163	-1.36886	0.56621	0.408148	-0.03373	-0.01574	0.001392	-0.03282
-0.46042	0.956657	0.574489	-0.28286	0.054551	-0.07059	0.023165	0.119043
-0.66519	1.028897	0.765497	-0.27148	0.158634	0.017976	0.06354	0.176903
-1.67514	0.87093	1.259652	3.035217	-0.34689	0.340888	0.179362	0.233459
0.36633	-0.50455	-0.33665	-0.6536	0.055098	0.393583	0.288392	0.927868
-0.02462	-0.43128	-0.00513	-0.00431	-0.04612	-0.20231	-0.19726	-0.27521

0.153092	-0.23475	-0.98576	-0.42282	0.263631	0.060319	-1.11357	-1.25695
0.145004	-0.17851	-0.11793	-0.09077	0.080737	0.097974	0.082656	-0.16237
0.045358	0.085788	-0.18157	-0.14968	0.03723	-0.42814	-0.60339	0.272412
0.26282	-0.25309	-0.16593	-0.07795	0.106687	0.005631	-0.04429	-0.11247
-0.21975	0.075456	0.018066	0.064638	0.028902	0.061766	-0.00929	-0.10038
0.012262	-0.05259	0.000687	0.184746	-0.03645	0.141669	0.200046	-0.06361
-0.04569	-0.00608	-0.02805	0.049776	0.002043	0.160566	0.211681	-0.06683
0.093463	-2.2807	0.008045	1.28981	-0.99265	0.832372	-0.51714	-2.44368
-1.54025	0.131775	-1.16587	-2.30575	0.121295	-0.86996	0.712183	-0.14682
0.268222	-0.27245	-0.16203	-0.09826	0.113133	0.00763	-0.0108	-0.11549
-0.08556	-0.03612	-0.02118	-0.01622	0.01284	-0.01876	0.013612	-0.01809
0.146527	-0.17978	0.114311	-0.19166	0.111577	-0.0317	0.029222	0.091286
0.205514	0.457891	0.056146	0.189711	-0.30063	0.398788	-0.16563	0.101584
-0.02374	0.052512	0.021414	-0.00677	0.00961	0.025791	-0.04316	-0.00977
-0.03166	0.064361	0.019374	0.011485	-0.00382	0.07905	-0.01994	0.024745
-0.06263	-0.12352	-0.03014	-0.0697	0.057388	-0.12623	0.064509	-0.03331
-0.01764	-0.05478	-0.02815	-0.04879	0.008547	-0.27139	0.064239	-0.02573
-0.02394	0.068375	-0.00623	-0.07852	0.065632	-0.27626	0.076251	-0.00815
-0.02457	0.055729	-0.02492	0.188487	0.344249	-0.4265	0.204472	-0.36668
-0.12854	0.078628	-0.01588	-0.03625	-0.03125	0.090752	-0.14736	0.085293
-0.08251	0.023081	-0.02282	-0.01289	-0.03774	0.015703	0.044594	0.14307
0.146737	0.097958	0.072015	-0.07619	-0.04275	0.061436	-0.06552	-0.13375
0.162442	0.104216	0.074473	-0.09043	-0.02655	0.035923	-0.06256	-0.18076
-0.55583	-3.34314	-3.53173	0.512504	-0.45322	-0.80259	0.860898	0.834917
-0.12668	0.028635	-0.01222	-0.03314	-0.08414	0.097722	0.115323	0.127821
-0.02785	0.038211	0.072612	-0.0034	-0.09908	0.098686	0.000984	0.006457

-0.21911	-0.1208	-0.12267	-0.19848	0.137261	-0.00879	0.054087	-0.03533
0.005398	-0.03082	0.091014	-0.03039	-0.00612	0.013275	-0.01447	0.006794
-0.13109	0.032458	-0.00696	0.123562	-0.02103	-0.08404	-0.01186	-0.00047
-0.03749	0.003809	0.176913	-0.0023	-0.01445	8.58E-05	-0.00055	0.007613
0.047531	0.049819	0.03394	-0.02055	0.023987	0.003689	-0.00475	-0.00099
0.004018	-0.01919	0.043685	-0.01523	-0.02473	0.048657	0.015654	-0.00861
0.004515	-0.03169	0.090561	-0.03491	-0.01244	0.035188	0.003218	-0.00514
-4.11546	2.286793	-0.79452	0.46845	-0.90705	-1.06202	0.106818	0.06151
-0.03348	0.002546	0.180782	-0.00472	-0.0156	0.001054	-0.00058	0.008003
0.048156	0.059805	-0.0358	0.025884	0.004993	0.01582	0.028077	-0.00075

0.037657	-0.00362	0.381661	0.152621	-0.0248	-0.20366	-0.1477	-0.01733
-0.00724	-0.00034	0.002899	0.013572	-0.00729	-0.02671	-0.0047	0.0429
-0.00015	0.00199	-0.10171	0.030897	0.005475	0.078736	0.002423	0.117059
-0.00758	-0.00072	-0.01263	0.003918	0.024816	-0.00709	0.003982	0.038784
0.008924	-0.00132	-0.00443	0.021693	0.010594	-0.01045	0.017029	-0.00907
0.004519	0.002258	0.035904	-0.01284	-0.01069	-0.03319	-0.02008	-0.06247
-0.00048	0.002131	0.032971	-0.00513	-0.02122	-0.03591	0.000983	-0.01544
-0.00977	0.074983	-0.24582	-0.51039	0.514885	0.062606	0.184317	0.926697
-0.00769	-0.00078	-0.01277	0.005792	0.025346	-0.0083	0.004302	0.036923
0.0057	-0.00593	0.002877	0.005908	0.015957	-0.01358	-0.03957	-0.0483

-0.06009	-0.02573	-0.15096	-0.13476	0.173026	0.183017	0.18029	-0.02283
0.025217	-0.03159	-0.00113	-0.01325	-0.00474	0.002313	0.034295	0.021302
-0.07748	-0.01535	-0.00569	0.015461	-0.0455	-0.00998	-0.03547	0.01522
0.045992	-0.02185	0.01106	-0.0075	-0.01009	-0.02555	0.02961	0.037248
-0.00978	-0.01027	-0.02201	-0.01259	0.004296	0.011988	-0.00485	0.007953
0.022331	0.011637	0.016669	0.006118	0.009504	0.002145	0.011669	-0.013
0.017217	-0.00671	0.008083	-0.00452	0.015392	0.009748	0.027902	0.001388
4.790084	1.381317	2.174723	5.862019	-1.84729	2.715497	4.720275	-3.13074
0.043542	-0.02406	0.010331	-0.00919	-0.00868	-0.02345	0.030458	0.037634
-0.03103	0.013779	-0.01511	-0.0112	0.0024	-0.01357	-0.02411	-0.00469

-0.04117	0.32673	0.185074	0.078244	-0.13912	0.129997	0.013557	0.065301
0.089144	-0.00487	0.086259	0.030044	-0.03967	0.024926	-0.01046	0.008455
0.024208	0.017749	-0.02315	0.033763	-0.04966	0.015726	0.035364	-0.01856
0.088557	-0.01921	0.113988	0.047821	-0.08247	0.032677	0.001496	0.001561
0.00921	0.014574	-0.01343	-0.00321	0.013434	0.000731	-0.00789	0.000474
0.000387	-0.01399	0.022538	0.001467	-0.00207	0.006419	-0.00379	0.00828
0.042752	-0.0187	0.050749	0.012703	-0.00411	0.009233	-0.01652	0.012102
-2.81353	-2.9511	-0.48083	-0.15682	-0.50776	2.736617	0.304367	2.813478
0.091431	-0.01901	0.114039	0.048765	-0.08268	0.033812	0.001348	0.002249
0.017142	0.027395	-0.11264	-0.02454	0.036674	0.004331	-0.01253	-0.00844

-0.01261	0.038651	-0.28163	-0.01561	0.01736	0.388389	0.310078	-0.02983
0.0076	-0.07861	-0.00363	-0.00235	0.093733	0.029018	-0.35292	-0.11713
-0.01438	0.011729	0.017141	0.055743	0.116237	-0.03273	-0.20904	-0.00894
-0.00668	-0.11259	-7.22E-05	0.016875	0.171387	0.031702	-0.51221	-0.15051
0.003255	0.001342	0.002986	-0.00921	-0.01358	0.008073	-0.02173	-0.01084
-0.00076	-0.00218	-0.00675	-0.00367	0.000275	0.014535	0.021011	-0.00393
0.013505	-0.02175	-0.00887	-0.02367	-0.00996	0.037569	-0.07926	-0.0244
-0.81805	-4.28553	0.676899	-2.15346	-3.02031	-0.29966	2.457507	0.497404
-0.00681	-0.1121	0.000444	0.015691	0.170377	0.033434	-0.51425	-0.15186
-0.00941	0.037057	0.004995	-0.00193	-0.00671	-0.0523	0.216915	-0.02044

-0.03452	-0.16805	-2.01771	0.07269	-1.15699	-1.31997	-0.3013	0.648507	DOS	DOS
-0.03231	-0.58415	0.235833	0.311408	-0.78918	0.318157	0.247363	1.591094	U2R	U2R
0.072983	0.497691	-0.0434	-0.31022	0.35957	-0.15293	-0.18622	0.305336	DOS	DOS
-0.0592	-0.81376	0.262462	0.239644	-1.04296	0.264488	0.574554	-0.16736	PROBE	PROBE
0.004634	0.040834	-0.01029	-0.05564	-0.06109	0.003275	0.002845	-0.08558	DOS	DOS
-0.02859	-0.14757	-0.00588	0.056306	0.113992	-0.0383	0.031037	-0.11301	normal	normal
-0.02309	-0.14755	0.03957	0.054862	-0.15854	0.002336	0.038658	-0.36615	U2R	U2R
0.028088	1.125286	-0.06559	0.115585	1.159934	-0.1756	-0.32748	1.88739	normal	normal
-0.06111	-0.82355	0.258799	0.24499	-1.05705	0.266951	0.578012	-0.19794	R2L	R2L
-0.03331	-0.27421	-0.14254	0.038104	0.17542	0.019401	-0.09956	0.567177	normal	normal

To identify some errors, results from record number 285 to 295 are presented

13945	-2.10323	0.177106	2.059507	2.392953	1.219139	1.178649	-0.87307
234566	-2.63189	0.10808	0.392037	1.252919	0.210205	-1.1163	0.047431
113719	-2.60087	0.087072	0.460003	1.157997	0.2432	-1.14781	0.101184
79142	-2.36426	-0.20152	0.872232	-0.58232	0.227524	-2.07327	2.230025
58257	5.23576	3.711878	-0.57757	-0.24909	-0.97075	-0.98179	1.21632
111248	4.927342	3.622376	-0.63448	-0.51914	-1.55735	-1.42311	1.549533
21558	-2.09045	-0.31265	0.589255	-1.14842	-0.00693	-1.90453	2.28428
171535	-2.55955	0.040225	0.581654	1.023269	0.301822	-1.21703	0.201799
306268	4.909196	-4.62792	-0.20931	1.158004	0.191078	0.472318	-0.3153
297121	-2.09538	0.172523	2.066069	2.38133	1.213815	1.151924	-0.89299
221242	-1.27312	-0.1049	2.78226	-2.53718	-3.66848	1.172802	-0.84484

-1.50167	-0.60677	-0.18494	-0.50834	-0.15621	0.230596	0.059001	0.275354
1.860407	-0.37148	1.198877	0.780212	-0.19057	0.135332	-0.00753	0.131932
1.811502	-0.35504	1.215509	0.71424	-0.20833	0.145221	-0.01773	0.121028
-0.84962	0.2771	-1.13968	-0.8941	-0.0026	-0.24044	0.082192	-0.01317
-0.61331	-1.22902	0.826633	0.358493	-0.79506	-6.58068	1.298993	-0.82874
-0.28996	-1.10421	0.615719	1.295456	3.288967	-0.86332	0.175222	0.596372
-1.05749	0.448669	-1.61048	-1.12872	0.063899	-0.08816	-0.00583	-0.05512
1.732695	-0.31427	1.229674	0.611214	-0.24025	0.153375	-0.03291	0.100905
-1.02259	0.967952	-0.25511	0.33835	-1.8807	1.946544	-1.03999	0.542577
-1.52594	-0.60724	-0.19659	-0.51279	-0.15417	0.2319	0.054434	0.264388
0.690165	1.417019	0.466252	-0.91575	0.425308	0.484023	-0.13049	-0.01393

0.091237	-0.20453	0.556766	-0.23823	-0.44241	0.155492	-0.62161	-0.98656
0.276095	-0.09949	0.062849	-0.03937	0.15122	-0.03137	0.151212	0.181942
0.247267	-0.09576	0.030582	-0.02922	0.147256	-0.03284	0.143178	0.185126
0.01671	0.200535	-0.19264	0.405194	-0.85656	0.226949	0.106451	0.055293
-2.28912	-2.15418	1.590885	-0.0194	0.598815	-0.87003	-0.31953	-0.46069
1.148244	0.140647	-1.12464	-0.58695	0.741788	-1.13195	-0.82095	-2.19884
-0.14802	0.200764	-0.29888	0.437731	-0.61265	0.16812	0.068047	0.057062
0.195806	-0.0912	-0.02506	-0.01031	0.135388	-0.03368	0.129121	0.190595
0.316558	-0.76748	-0.79476	-0.50943	0.32329	0.374875	-0.53143	0.172729
-0.07364	0.16167	0.098624	0.058289	-0.08479	-0.01178	0.026981	-0.0708
-0.06621	0.17253	0.109628	0.066755	-0.08537	-0.01425	0.028553	-0.06837
-0.04723	-0.12218	0.018766	0.010938	0.056303	-0.14263	0.194158	0.030021
0.555277	0.927152	-0.63031	-0.29332	0.760726	1.804878	-1.672	-0.82098
1.956799	-0.36339	0.422216	1.741814	-0.27967	0.092756	-0.81731	-0.58228
-0.0403	-0.07249	0.04062	0.019906	0.046745	-0.10431	0.160923	0.024785
-0.05374	0.190226	0.130364	0.082443	-0.08615	-0.02126	0.03364	-0.06198
-2.23623	3.35377	0.323412	1.98849	0.075058	0.749068	0.353691	-2.84132
0.314358	-0.7769	-0.80009	-0.51165	0.308792	0.377526	-0.53295	0.162437
-0.00607	-0.09494	-0.00826	0.018335	-0.0039	0.111338	-0.02215	-0.02019

0.610441	0.332633	0.751681	0.150465	0.331131	-0.46933	0.660542	-0.27342
-0.17812	-0.09881	-0.15844	-0.04648	-0.05858	0.055503	-0.13444	0.056302
-0.19861	-0.09814	-0.16171	-0.05025	-0.0577	0.060671	-0.14139	0.057368
0.173275	0.056165	0.089285	0.023818	-0.01155	0.028444	0.052717	-0.01261
-1.00108	-0.80749	-1.16982	-0.8124	0.776668	-0.31765	-0.41004	-0.35623
-2.69629	1.220407	-2.80272	3.507898	0.293266	-0.62336	0.303718	-0.12946
0.139032	0.025541	0.078287	0.001978	-0.01408	0.044442	0.044779	-0.00308
-0.23492	-0.09621	-0.16661	-0.05628	-0.05591	0.070201	-0.15398	0.059336
-0.30498	-2.67633	-1.54451	-1.07761	-2.89343	-4.3108	2.830411	-0.82815
0.608178	0.328718	0.752463	0.147098	0.325594	-0.4702	0.658788	-0.27179
-0.01307	-0.06601	0.049236	-0.07588	-0.09182	-0.01823	-0.00161	-0.00098

0.163669	-0.18682	0.052112	-0.01185	0.187399	-0.02485	-0.13988	-0.01893
-0.05443	0.004837	-0.0416	0.039858	-0.02802	-0.02065	0.037053	0.00824
-0.05379	0.011076	-0.03961	0.039529	-0.02817	-0.02122	0.037004	0.008748
0.070066	0.167171	0.094075	-0.07026	-0.02283	0.018942	0.009735	-0.00353
-0.10743	0.537104	-0.68046	-0.52695	-0.54985	-0.18905	-0.36877	0.037238
-0.82223	-6.2511	9.118171	0.017024	-2.19525	-3.31419	-5.86201	1.00617
0.057556	0.193555	0.117575	-0.0392	-0.03041	0.010829	0.023682	-0.01539
-0.05227	0.023023	-0.03565	0.038215	-0.02829	-0.02189	0.036465	0.01017
2.365137	3.799115	6.593063	2.184718	3.623008	-2.30523	-1.3108	-0.23073
0.163228	-0.18849	0.051097	-0.01205	0.186976	-0.02449	-0.14012	-0.01844
-0.0606	-0.00589	-0.02981	0.055667	-0.05188	0.034172	-0.03363	-0.01693

0.015341	-0.01031	-0.00044	-0.16451	0.033341	0.019526	0.122754	0.017018
-0.00677	0.002057	0.001825	0.034804	-0.00274	-0.01721	-0.02851	-0.00168
-0.00617	0.001386	0.001427	0.03501	-0.00022	-0.02238	-0.02912	-0.00385
0.005886	0.012859	-0.00113	-0.01314	0.002859	0.0345	-0.0432	-0.01258
0.000183	0.044791	0.001912	0.032271	0.456743	-0.50593	0.497977	0.405524
0.034888	0.075328	0.157105	-0.25239	2.504697	-2.72507	2.15486	0.003834
0.000917	0.013517	-0.00191	-0.01261	0.023844	0.005256	-0.05136	-0.01735
-0.00475	9.69E-05	0.000712	0.035249	0.00377	-0.03114	-0.03007	-0.00748
0.145117	-0.19679	0.06525	-0.93432	-1.46739	0.92577	-3.87809	1.558001
0.015205	-0.01013	-0.00046	-0.16349	0.033836	0.019536	0.123223	0.017196
0.002829	0.000481	0.005213	0.007212	0.006692	0.01764	-0.0759	-0.03816

0.215973	-0.12505	-0.03975	-0.0095	0.018266	-0.06163	-0.01623	-0.03104
-0.02462	0.022715	0.00508	0.007001	0.009094	0.010935	0.011336	0.020106
-0.02431	0.020852	0.003501	0.004466	0.008428	0.010536	0.012985	0.021158
-0.15784	0.00086	-0.00595	-0.02258	-0.01783	0.015962	0.006623	-0.01079
0.147676	0.112903	0.089722	-0.30027	0.086224	0.237236	0.324512	0.148228
-2.0984	-4.08055	4.032051	-1.02729	-2.43555	-0.56662	-0.56089	-1.43806
-0.16899	-0.00526	-0.01532	-0.03709	-0.01981	0.01351	0.020922	-0.01034
-0.02425	0.01746	0.000552	2.54E-05	0.007067	0.009887	0.015818	0.023213
4.501136	4.111309	0.347039	1.086104	0.235119	-0.34403	-2.25829	2.565339
0.216246	-0.12522	-0.03956	-0.00954	0.018065	-0.06127	-0.01581	-0.03083
0.009136	-0.01151	-0.00674	0.001412	0.026583	0.046356	0.039762	0.027143

0.029604	0.107687	0.010271	-0.0263	0.067469	-0.06831	0.031695	0.043259
-0.00567	0.013955	-0.01375	0.027062	0.0054	0.00956	0.002725	-0.01178
-0.0041	0.023727	-0.0141	0.022144	0.006072	0.013328	0.003239	-0.01404
0.003787	0.004793	0.028036	-0.05203	-0.0011	0.00044	0.006126	0.01001
0.168124	-0.30662	0.004617	-0.16555	-0.0486	0.501406	-0.27609	0.048862
1.637229	-0.32131	-0.59973	1.021982	0.174058	-0.01427	0.634604	0.273802
0.008696	0.018084	0.028426	-0.05114	0.002396	0.00925	0.006003	0.007355
-0.0012	0.042397	-0.01513	0.013195	0.007527	0.020048	0.00432	-0.01814
0.468443	-0.7272	-0.40087	1.371332	0.514365	-0.28446	0.720522	-1.59168
0.029466	0.106557	0.010833	-0.02579	0.06748	-0.06873	0.031812	0.043446
0.015309	-0.01554	-0.00594	0.040799	0.030405	0.018224	0.018242	0.048773

-0.02351	-0.01668	0.009195	0.024688	0.065615	0.170761	-0.05813	-0.32255
0.006145	0.011207	-0.00268	-0.00773	-0.01754	-0.02951	0.016725	0.001814
0.005456	0.014228	-0.00033	-0.00819	-0.02016	-0.03608	0.014637	0.012816
0.00021	-0.01497	0.044569	0.005618	0.01538	-0.02191	-0.00778	0.147473
-0.06001	0.391849	0.179115	-0.13489	-0.30492	-1.10713	-0.10563	1.336603
0.744756	0.034422	0.754027	0.012426	-0.00794	-0.12008	0.876843	0.598302
0.000796	-0.00094	0.05096	0.002897	0.010661	-0.05169	0.005677	0.151055
0.004208	0.019576	0.004028	-0.00903	-0.02493	-0.04777	0.010583	0.033033
-0.49607	-0.41443	-0.5042	0.179255	0.479736	1.652984	-0.55448	-1.14812
-0.02346	-0.01686	0.009005	0.023992	0.065593	0.171059	-0.05726	-0.32243
0.006872	-0.00567	0.014121	0.015603	-0.02491	-0.05274	0.024598	-0.02981

-0.03079	0.079522	0.55564	-0.01073	-0.22497	0.341536	-0.11896	-0.12891	0.00708	normal	normal
0.002539	-0.00788	-0.05171	0.050727	0.03004	-0.04354	-0.00839	0.01935	-0.021	R2L	R2L
0.000224	-0.00866	-0.06074	0.04721	0.03738	-0.05932	-0.00189	0.00828	0.007375	DOS	DOS
0.0258	-0.01978	-0.1158	-0.12727	0.153186	-0.07586	0.053775	0.033606	-0.04925	DOS	DOS
0.716844	0.87198	6.61661	0.89756	-4.25718	-6.36996	1.39164	0.387119	1.80654	Normal	Normal
0.**307746**	0.118571	1.**903448**	-0.09946	0.**198819**	1.**496094**	-0.**11647**	-0.57824	1.**750175**	**Normal**	**U2R**
0.023333	-0.01152	-0.04006	-0.11395	0.11133	-0.14312	0.053542	-0.03307	0.006546	DOS	DOS
-0.00405	-0.01095	-0.08156	0.038541	0.05426	-0.08706	0.010178	-0.00855	0.051269	PROBE	PROBE
0.373686	0.360647	2.960279	-0.21389	-0.53043	1.661562	0.056836	-0.42663	-0.15791	U2R	U2R
-0.03059	0.079575	0.556624	-0.0144	-0.22765	0.34051	-0.12246	-0.12739	0.006204	U2R	U2R
0.06521	-0.00409	0.142647	0.229809	-0.08622	-0.2365	-0.04055	0.311207	-1.34343	R2L	R2L

Please observe that record number 290 (Bolden) was misclassified, with Normal predicted as U2R

4.4 Confusion Matrix

Confusion matrix depicts the relationship between the actual (or expected) values and the predicted values. The confusion matrices depicted in this work are multi-class confusion matrices.

K-Nearest Neighbor (KNN)

Figure 4.1 depicts the confusion matrix of KNN. These predictions were evaluated using a custom `evaluate model` function, with performance visualized through a confusion matrix and a detailed classification report, highlighting metrics such as precision, recall, and F1-score.

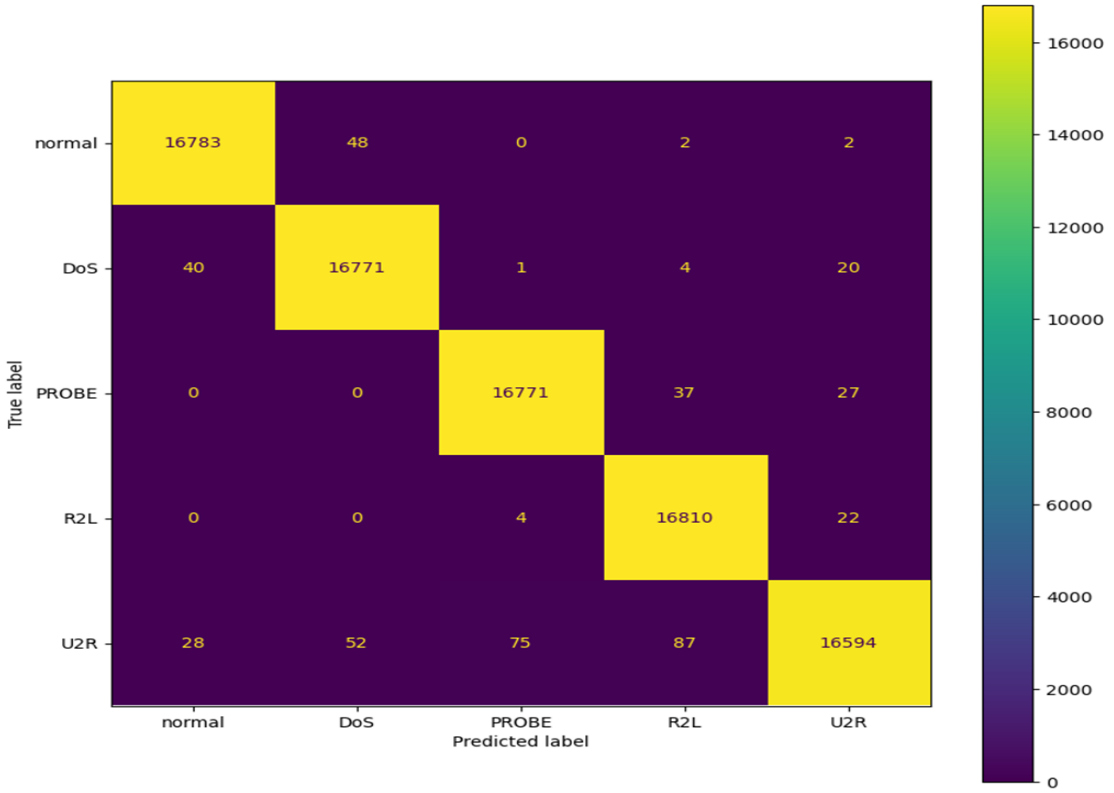


Figure 4.1: Confusion Matrix for KNN
Source; Researcher (20240

.

From Figure 4.1, KNN predicted 16783 TP records as normal; 16771 TP records as DOS, 16771 TP records as PROBE, 16610 TP records as R2L, and 16594 TP records as U2R. Observe that all TP values lie in the main diagonal of the confusion matrix. However, the model misclassified 48 normal records as DOS, 2 normal records as R2L and U2R respectively. KNN also misclassified 40 DOS records as normal, one DOS record as PROBE, 4 DOS records as R2L, and 20 DOS records as U2R. Similarly, the model misclassified 37 PROBE records as R2L and 27 PROBE records as U2R. It also misclassified 4 R2L records as PROBE and 22 U2R records as R2L. Finally, the model also misclassified 28 normal records as U2R, 52 DOS records as U2R, 75 PROBE records as U2R and 87 R2L records as U2R. . These predictions were evaluated using a custom `evaluate model` function, with performance visualized through cross validation and a detailed classification report, highlighting metrics such as accuracy,

precision, recall, F1-score, and area under the receiver operating curve AUC-ROC) curve, true positive rate (TPR), false positive rate (FPR), true negative rate (TNR), and false negative rate (FNR). In malware detection problem, TP refers to the number of normal executives predicted as normal; TN refers to the number of malicious records predicted as malware, FP refers to the number of true positive records predicted as malicious; while FN refers to malicious records predicted as normal records.

TPR is defined as the number of true positives divided by the total number of true positives divided by total number of malicious executive files, as depicted in Equation 4.1.

$$\text{TPR} = \frac{TP}{TP+ FN} \qquad \text{Equation 4.1}$$

FPR is defined as the number of false positive divided by total number of benign executive files, such as depicted in Equation 4.2.

$$\text{FPR} = \frac{FP}{FP+ TN} \qquad \text{Equation 4.2}$$

Precision is defined as true positive numbers divided by the sum of positive number and false positive numbers, as depicted in Equation 4.3

$$\text{Precision} = \frac{TP}{TP+ FP} \qquad \text{Equation 4.3}$$

Recall is defined as true positive numbers divided buy the sum of true positive numbers and false positive numbers, as depicted in Equation 4.4

$$\text{Recall} = \frac{TP}{TP+ FN} \qquad \text{Equation 4.4}$$

Accuracy is defined as the sum of the true positive numbers and true negative numbers divided by the total number of instances, as depicted in Equation

$$\text{Accuracy} = \frac{TP+ TN}{TP+ FP+ FN+ TN} \qquad \text{Equation 4.5}$$

The KNN model demonstrated strong performance with an accuracy of 0.98816, precision of 0.988158, recall of 0.988170, F1-Score of 0.988152, and an overall AUC-ROC score of 0.99740. Table 4.2 depicts KNN performance metrics

Table 4.2: KNN Performance Metrics

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
KNN	0.98816	0.988158	0.988170	0.988152	0.9991

Source: Researcher (2024)

The AUC-ROC scores for individual attack classes were also impressive, with normal at 0.9980, DOS at 0.9974, PROBE at 0.9982, R2L at 0.9986, and U2R at 0.9949.

The confusion matrix further highlights the model's effectiveness, showing minimal misclassifications across all classes. The AUC-ROC scores for individual attack classes were also impressive, with normal at 0.9980, DOS at 0.9974, PROBE at 0.9982, R2L at 0.9986, and U2R at 0.9949. Figure 4.2 depicts the AUC-ROC plot. Specifically, the AUC curve establishes the relationship between false negatives and false positives. While ROC curve is obtained by plotting the TPR against FPR (Santos *et al.,* 2013)

These results indicate that the KNN model is highly capable of accurately detecting and distinguishing between different types of network attacks, making it a reliable choice for network intrusion detection.

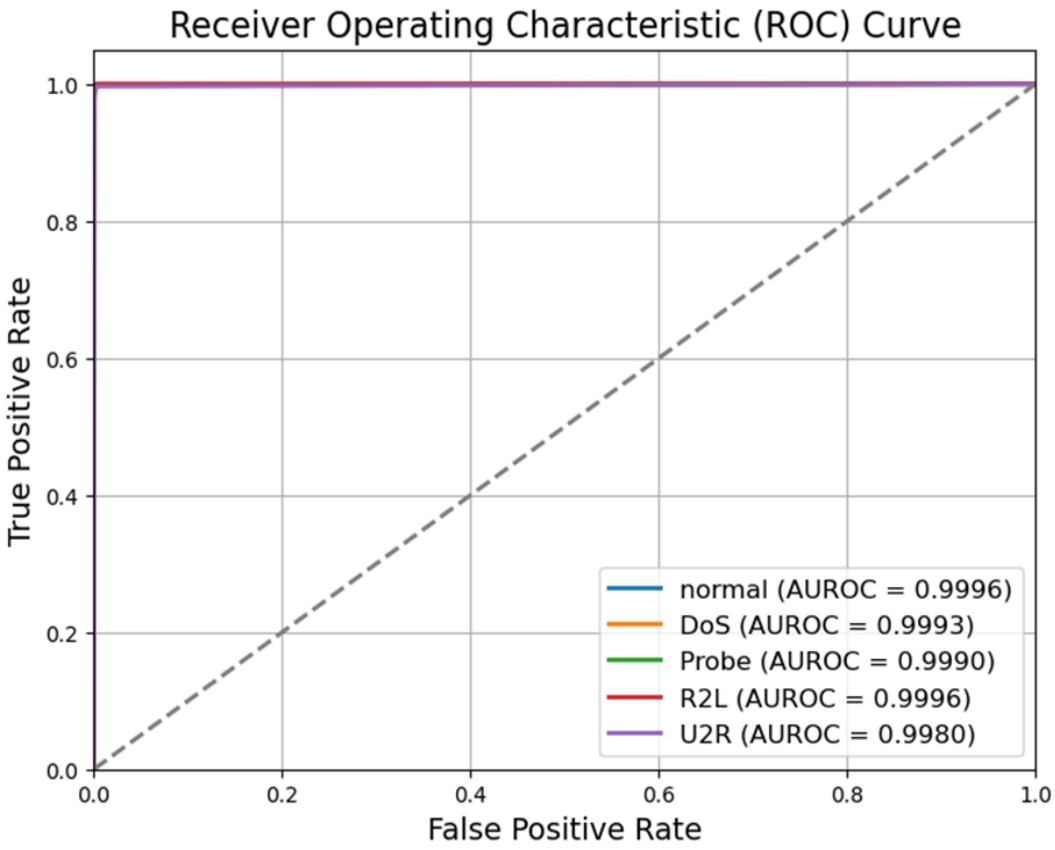


Figure 4.2: AUC-ROC Curve of KNN Classifier
Source: Researcher (2024)

Random Forest Classification

To train and evaluate a Random Forest model, it was initialized with 500 estimators to ensure robust and stable model performance by aggregating the predictions from multiple decision trees, thus reducing over fitting. Entropy was used as the criterion to measure the quality of splits, focusing on maximizing the information gain at each split. The model was trained on the training dataset (80%), and predictions were made on the test dataset (20%).

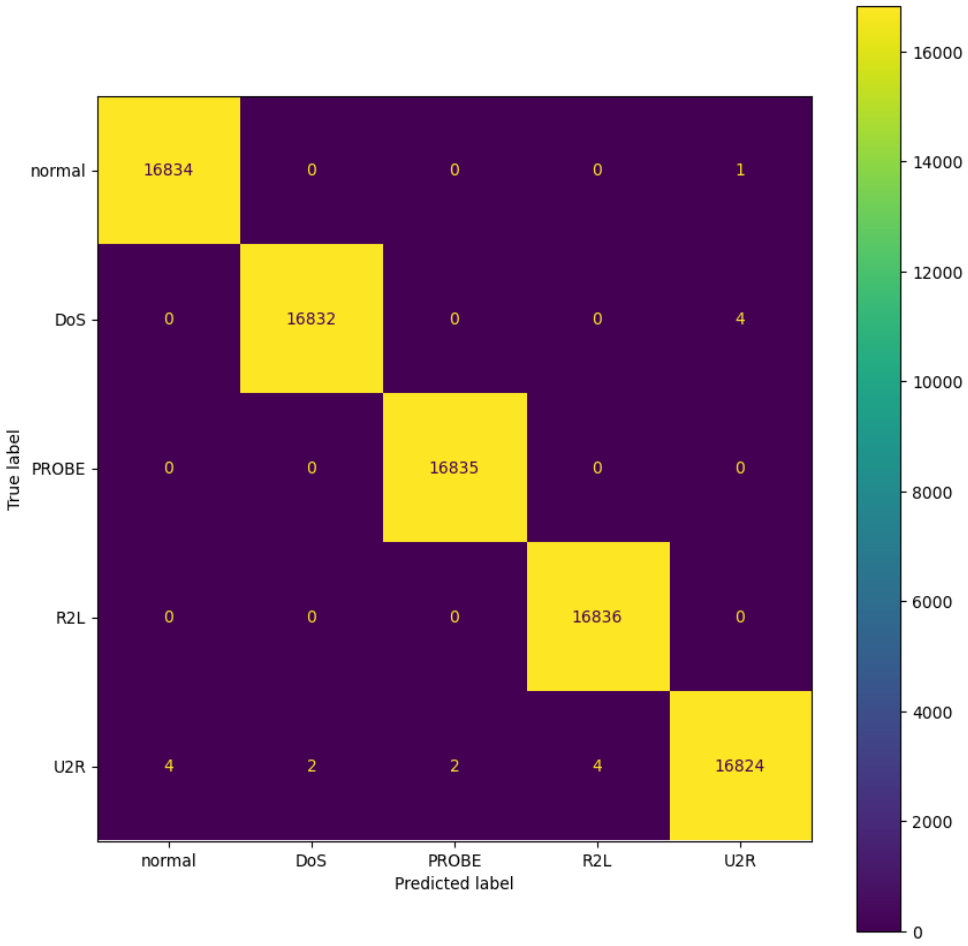


Figure 4.3: Confusion Matrix for Random Forest
Source: Researcher (2024)

Figure 4.3 shows that RF correctly predicted 16834 TP records as Normal, 16832 TP records as DOS, 16835 TP records as PROBE, 16836 TP records as R2L, and 16824 TP records as U2L with all the TP records displayed in the main diagonal of the confusion matrix. However, the model misclassified 1 normal record as U2R, 1 U2R record as DOS, 4 Normal records U2R, 2 DOS records as U2R, 2 PROBE records as U2R and 4 R2L records as U2R respectively. To assess the model's performance, a custom evaluate_model` function was used, and confusion matrix used to compare the predictions with the actual or expected values. Table 4.3 depicts the performance metrics for RF.

Table 4.3: Performance metrics for RF Classifier

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Random Forest Classifier	0.99932	0.999322	0.988170	0.999319	1.0

Source: Researcher (2024)

The Random Forest Classifier demonstrated excellent performance metrics, achieving an accuracy of 0.99932, precision of 0.999322, recall of 0.988170, F1-Score of 0.999319, and a perfect overall ROC-AUC score of 1.0. The AUC-ROC scores for each attack class were outstanding with 1.0 in all the classes, as depicted in Figure 4.4 Specifically, the AUC curve establishes the relationship between false negatives and false positives. While ROC curve is obtained by plotting the TPR against FPR (Santos *et al.,* 2013)

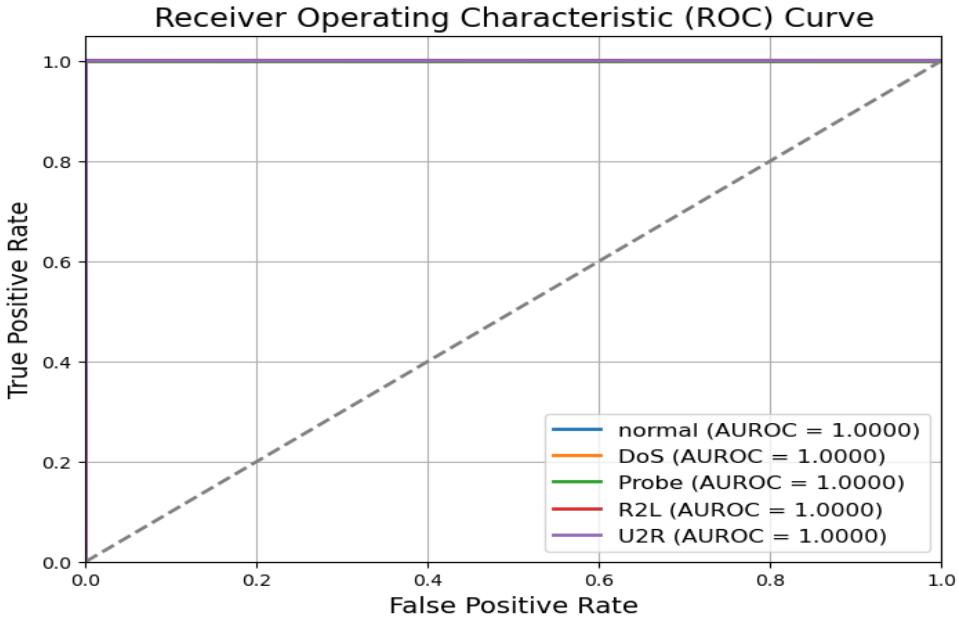


Figure 4.4: AUC-ROC operating curve of Random Forest

Source: Researcher (2024)

Despite these high scores, there were still some misclassifications evident in the confusion matrix. These misclassifications highlight that while the Random Forest

Classifier is highly effective at distinguishing between various types of network attacks, some errors do persist, particularly with the 'DOS' and 'U2R' attack classes.

Naïve Bayes Classifier

Training and evaluating a Naive Bayes model, I utilized the Gaussian Naive Bayes classifier due to its effectiveness in handling continuous data and its assumption of a normal distribution for the features. The model was trained on the training dataset (80%), and predictions were made on the validation dataset (20%).

Figure 4.5 depicts the confusion matrix of Gaussian Naïve Bayes classifier.

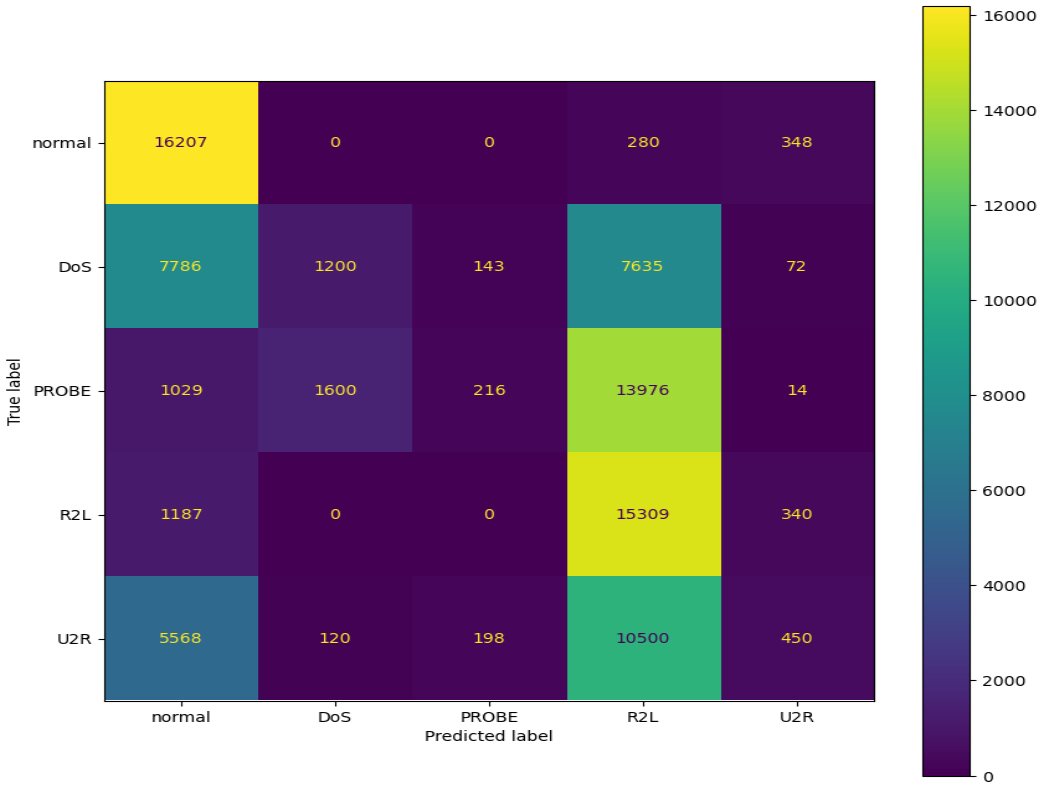


Figure 4.5: Confusion Matrix of Naïve Bayes classifier
Source: Researcher (2024)

Figure 4.5 depicts that Naïve Bayes classifier correctly predicted 16207 TP records of Normal type, 1200 TP records of DOS, 216 TP records of PROBE attack type, 15309 TP records of R2L attack type, and 450 TP records of U2R attack type. However, the model misclassified 7700 DOS records as Normal type, 1020 records of PROBE as

Normal, 1187 records of R2L type as Normal and 5568 records of U2R records as Normal. Similarly, the model misclassified 1000 records of PROBE type as DOS and 120 records of U2R as DOS. The model also misclassified 143 records of DOS as PROBE and 198 records of U2R as PROBE attack type. The model also misclassified 280 records of Normal type as R2L, 7635 records of DOS types as R2L, 13973 records of PROBE attack types as R2L, 1050 records of U2R records as R2L. Again, the model misclassified 148 records of Normal type as U2R, 72 records of DOS attack type as U2R, 14 records of PROBE attack type as U2R and 340 records of R2L attack type as U2R attack type.

To assess the model's performance, custom evaluate model function was used and the parameters derived are displayed in Table 4.4.

Table 4.4: Naïve Bayes Model Performance Evaluation results

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Naive Bayes Classifier	0.39624	0.410736	0.396438	0.268998	0.7419

The Naive Bayes Classifier displayed significantly lower performance metrics compared to other models, with an accuracy of 0.39624, precision of 0.410736, recall of 0.396438, F1-Score of 0.268998, and an overall AUC-ROC score of 0.7419. The AUC-ROC scores for individual attack classes were: class 0 (Normal) at 0.9032, class 1 (DoS) at 0.4746, class 2 (PROBE) at 0.6181, class 3 (R2L) at 0.8777, and class 4 (U2R) at 0.8448 as depicted in Figure 4.7. Figure 4.7 compares the false positive rate score with that true positive rate and derived the area under the curve depicted. Specifically, the AUC curve establishes the relationship between false negatives and

false positives. While ROC curve is obtained by plotting the TPR against FPR (Santos *et al.,* 2013).

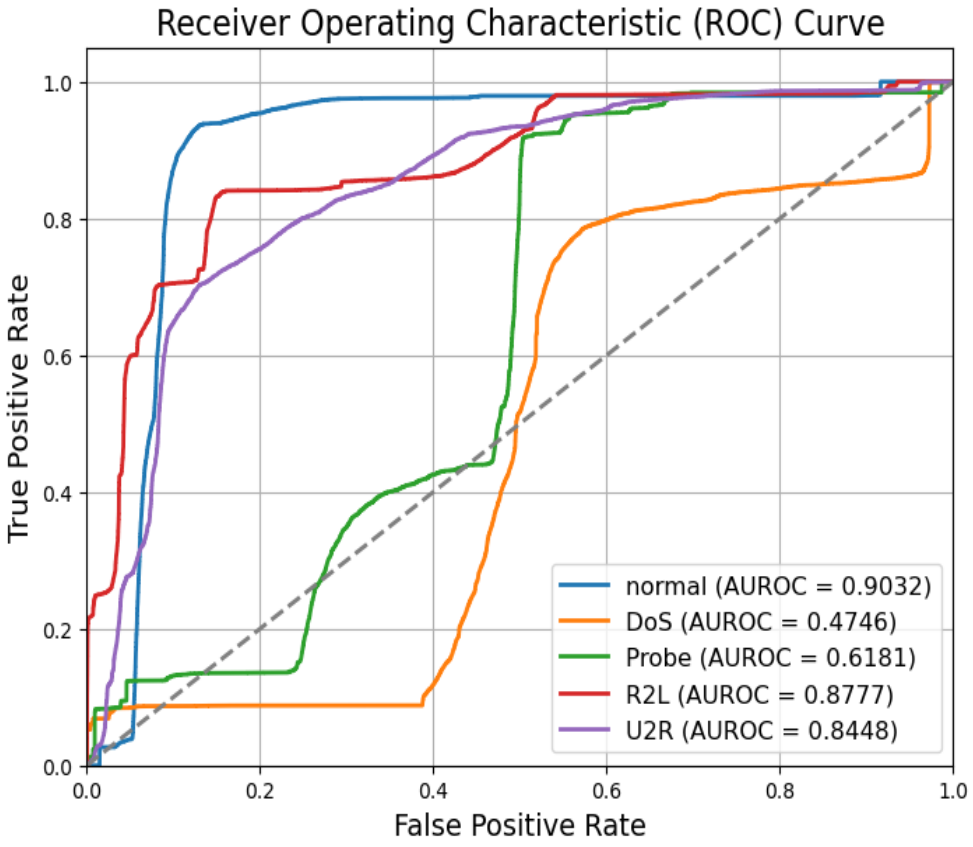


Figure 4.6: Comparison of FPR and TPR of Naïve Bayes Classifier
Source: Researcher (2024)

The AUC-ROC curves highlight the classifier's challenges in correctly identifying the different types of network attacks. The high number of misclassifications in Figure 4.5, especially among 'DoS', 'PROBE', and 'U2R' attacks, suggest that the Naive Bayes Classifier struggled with the complexity of the dataset and the variability within attack classes. These results indicate that, while Naive Bayes provides some insight, it may not be the best choice for accurately classifying network intrusions in this context.

Decision Tree Classifier

To train and evaluate a Decision Tree (DT) model, entropy was used to initialize the it as the criterion to measure the quality of splits, focusing on maximizing the information gain at each split. The model was trained on the training dataset (80%), and predictions were made on the test dataset (20%). Figure 4.8 depicts confusion matrix of DT.

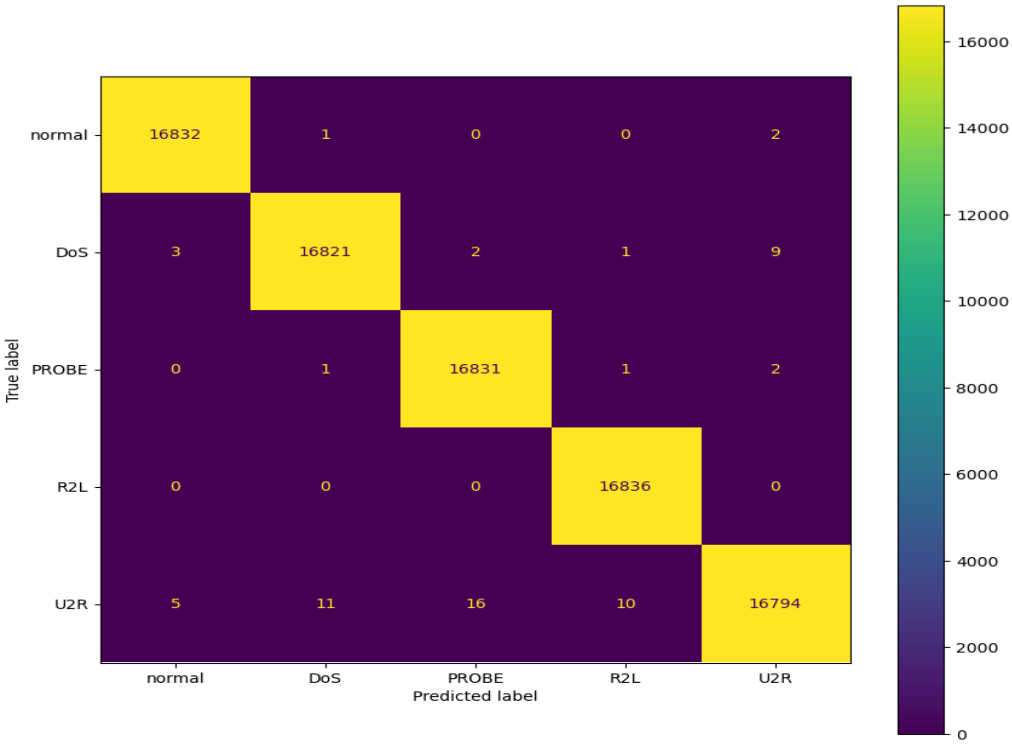


Figure 4.7: Confusion Matrix of Decision Tree Classifier.
Source: Researcher (2024)

The confusion matrix depicts 16832 TP records of Normal type, 16821 TP records of DOS, 16831 TP records of PROBE attack type, 16836 TP records of R2L attack type, and 10794 TP records of U2R attack type. The model, however, misclassified 3 records of DOS as Normal type and 5 records of U2R as Normal type. It also misclassified 1 record of Normal type as DOS attack type, 1 record of PROBE attack as DOS, and 11 records of U2R attack type as DOS attack type. Similarly, the model misclassified, the

model misclassified 2 records of DOS as PROBE attack type and 10 records of U2R attack as PROBE attack type. The model also misclassified 1 record of DOS attack type as R2L, 1 record of PROBE attack type as R2L, and 10 records of U2R attack type as R2L. In the same vein, the model misclassified 2 records of Normal type as U2R, 9 records of DOS attack type as U2R and 2 records of PROBE attack type as U2R. The low number of false positives and false negatives across all classes further underscores the model's reliability and accuracy in classifying network intrusion types, making it a robust choice for this task.

To assess the model's performance, k-fold cross validation, with k=10 was used, paving the way for splitting the dataset into 10 different sets, with 90% used to train and the remaining 10% used to assess the model. The resulting performance parameters are recorded in Table 4.5 as accuracy, precision, recall, F1-score and AUC of ROC

Table 4.5: Performance Metrics of Decision Tree

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Decision Tree Classifier	0.99804	0.410736	0.998042	0.998041	0.9988

The Decision Tree Classifier demonstrated impressive performance metrics, achieving an accuracy of 0.99804, precision of 0.410736, recall of 0.998042, F1-Score of 0.998041 and an overall UAC-ROC score of 0.9988 as depicted in Table 4.6. The AUC-ROC scores for individual attack classes were: class 0 (Normal) at 0.9999, class 1 (DoS) at 0.9995, class 2 (PROBE) at 0.9997, class 3 (R2L) at 0.9999, and class 4 (U2R) at 0.9987 as depicted in Figure 4.9. Specifically, the AUC curve establishes the relationship between false negatives and false positives. While ROC curve is obtained by plotting the TPR against FPR (Santos *et al.,* 2013).

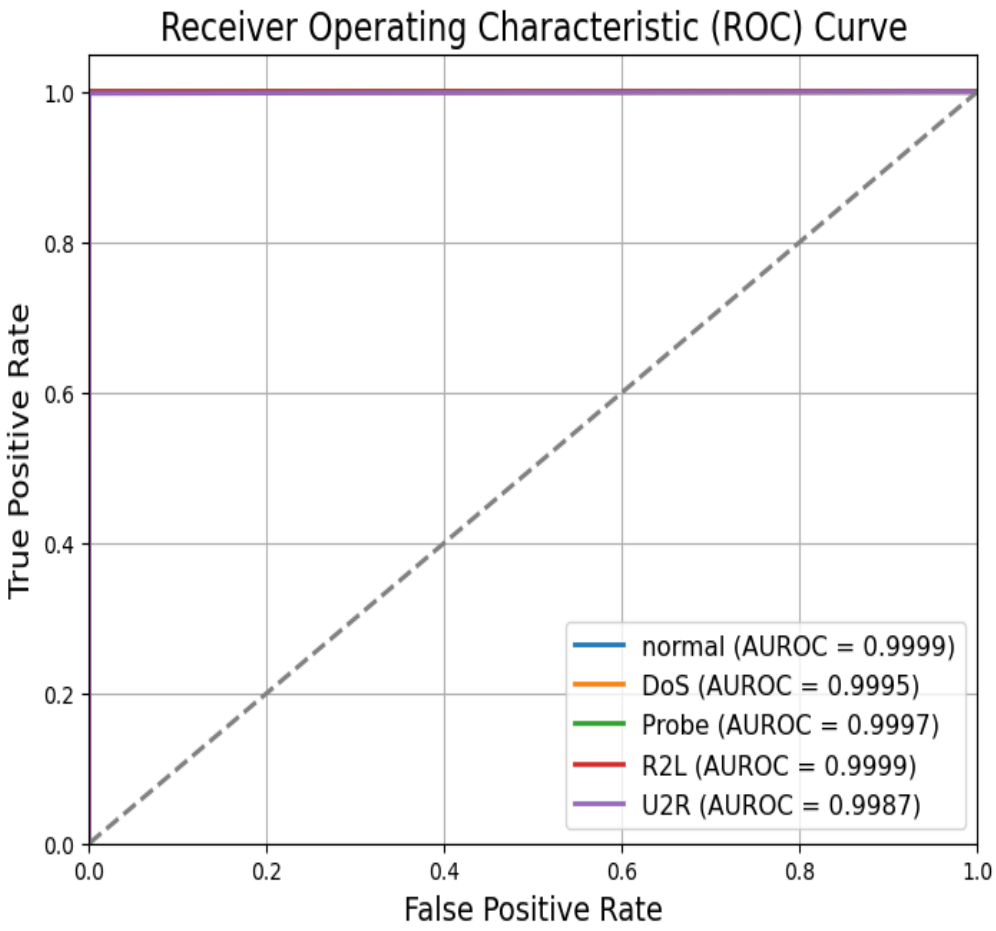


Figure 4.8: Area under the ROC Curve for Decision Tree
Source: Researcher (2024)

Logistic Regression (LR) Classifier

Training and evaluating the Logistic Regression model, the classifier was initialized with a maximum of 1000 iterations to ensure convergence. The model was trained on the training dataset, which comprised 80% of the total data, and predictions were made on the test dataset, which comprised the remaining 20%. Figure 4.9 depicts confusion matrix of LR

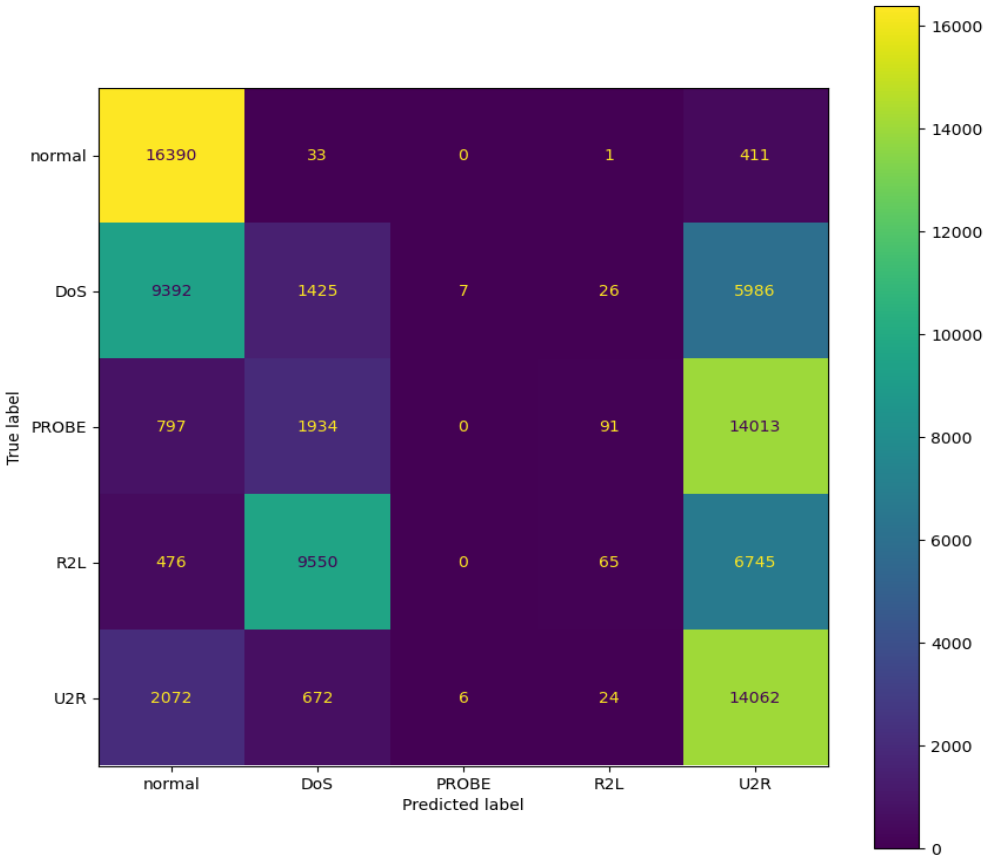


Figure 4.9: Confusion Matrix for Logistic Regression Classifier
Source: Researcher (2024)

From Figure 4.9, Logistic Regression correctly predicted 16390 TP records of Normal type, 1425 TP records of DOS, 6 5 TP records of R2L and 14062 TP records of U2R attack type. However, the model misclassified 9392 records of DOS as Normal, 797 records of PROBE as Normal, 476 records of R2L as Normal and 2072 records of U2R as Normal type. The model also misclassified 33 records of Normal as DOS, 1934 records of PROBE as DOS, 9550 records of R2L as DOS, and 672 records of U2R as DOS. Similarly, the model misclassified 7 records of DOS as PROBE attack type; and 1 record of Normal type as R2L, 26 records of DOS type as R2L, 91 records of PROBE type as R2L attack type, and 24 records of U2R attack types as R2L. Also, the model misclassified 411 records of Normal type as U2R attack type, 5986 records of DOS

attack type as U2R attack type, 1413 records of PROBE attack type as U2R and 6745 records of R2L attack type as U2R attack type.

To assess the model's performance, k-fold cross validation, with k=10 was used, paving the way for splitting the dataset into 10 different sets, with 90% used to train and the remaining 10% used to assess the model. The resulting performance parameters are recorded in Table 4.6 as accuracy, precision, recall, F1-score and AUC of ROC.

Table 4.6: Performance parameters for Logistic Regression

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC**
Logistic Regression	0.37945	0.26451	0.379460	0.259775	0.7421

In evaluating the Logistic Regression model using the NSL-KDD dataset, the performance metrics highlight the model's strengths and weaknesses. The model achieves a moderate overall accuracy of 37.945%, with low recall and precision resulting in an F1-Score of 25.9775% and a ROC of 0.7421.

The AUC-ROC scores for individual attack classes were: class 0 (Normal) at 0.9330, class 1 (DoS) at 0.2893, class 2 (PROBE) at 0.6684, class 3 (R2L) at 0.9354, and class 4 (U2R) at 0.8845 as depicted in Figure 4.10 Specifically, the AUC curve establishes the relationship between false negatives and false positives. While ROC curve is obtained by plotting the TPR against FPR (Santos *et al.,* 2013).

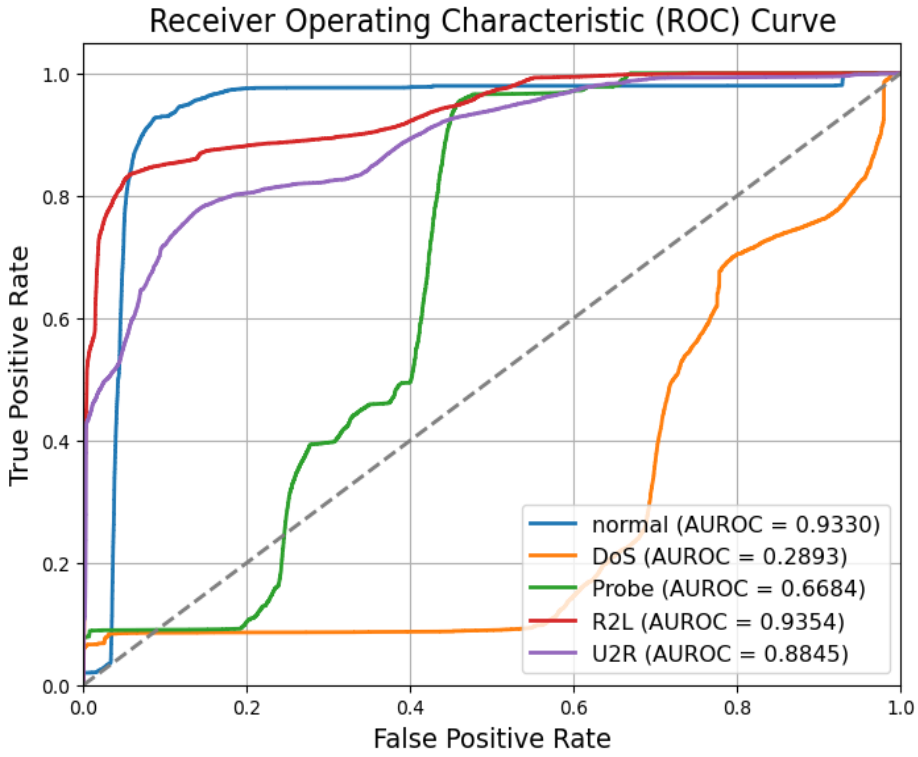


Figure 4.10: Area Under the ROC Curve of Logistic Regression
Source: Researcher (2024).

4.5 Ensemble Learning (Soft Vote) Classifier

The ensemble learning approach employed in this implementation leverages the strengths of multiple individual classifiers to enhance predictive performance. The voting classifier is the core of this strategy, combining five distinct models: K-Nearest Neighbors (KNN), Random Forest (RF), Naive Bayes (GNB), Decision Tree (DT) and Logistic Regression (LR). Soft voting is utilized, where the predicted class probabilities from each model are averaged, and the class with the highest average probability is selected. This ensemble method ensures a more balanced and robust prediction. The weights assigned to each model—slightly favoring KNN and Decision Tree—reflect their relative contributions to the ensemble, fine-tuning the overall performance. The process involves defining the ensemble model with these estimators, training it on the dataset, predicting outcomes using test dataset.

Figure 4.12 depicts the confusion matrix of the soft vote classifier, comparing the existing or expected labels with the predicted ones, and generating TP, FP, TN and FN values of the comparison.

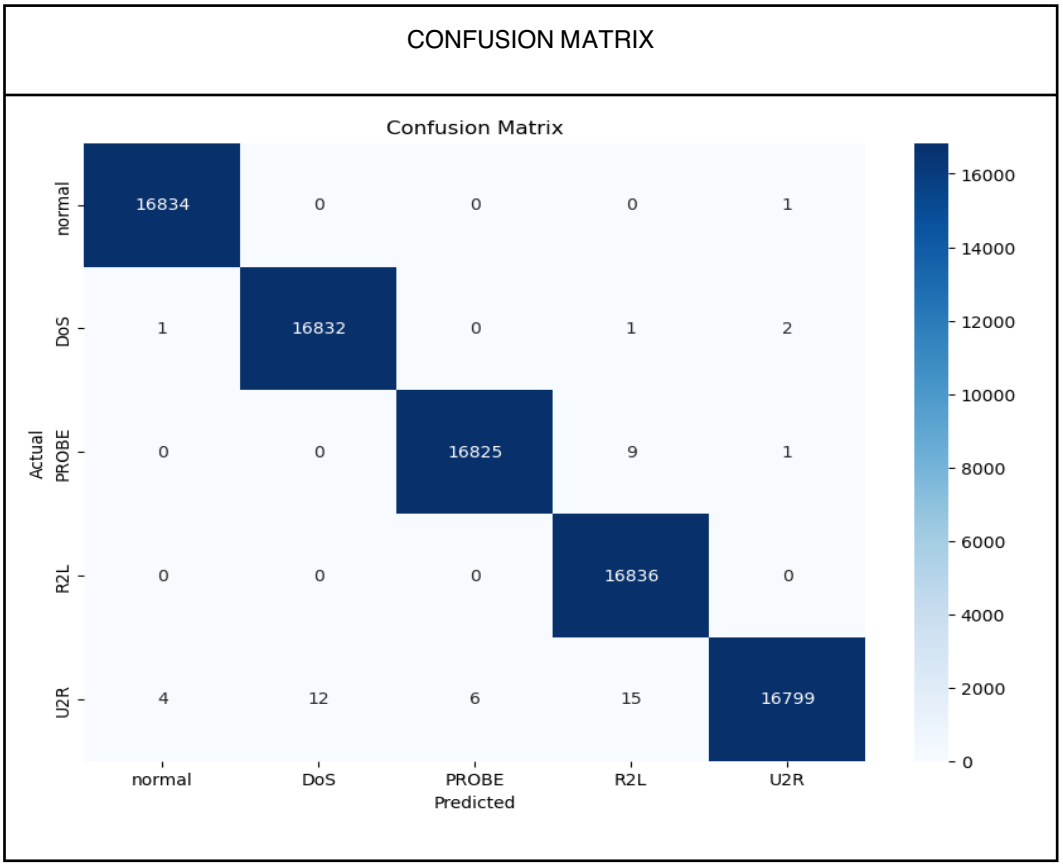


Figure 4.11: Confusion Matrix of Soft vote classifier
Source: Researcher (2024)

From Figure 4.12, soft vote classifier predicts 16834 TP records of Normal type, 16832 TP records of DOS, 16825 TP records of PROBE, 16836 TP records of R2L attack type, and 16799 TP records of U2R attack type. Although the misclassified records seem negligible, the following misclassifications were observed. 1 DOS record was misclassified as Normal and also 4 U2R records were misclassified as Normal. Similarly, 12 U2R records were misclassified as DOS attack type, 6 U2R records were also misclassified as PROBE attack type. Similarly, 1 DOS record was misclassified as R2L attack type, 9 PROBE records were misclassified as R2L attack type, and 15 U2R

records were misclassified as R2L attack type. The model also misclassified 1Normal record as U2R, 2 DOS records as U2R, and 1 PROBE record as U2R attack type.

Its performance was evaluated using k-fold cross validation with resulting metrics such as accuracy, precision, recall, F1-Score, and AUC-ROC as depicted in Table 4.7.

Table 4.7: Performance parameters of Soft Vote Classifier

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Ensemble Learning (Voting Classifier)	0.99788	0.99788	0.997889	0.99788	0.99996

Source: Researcher (2024)

The Voting Classifier demonstrates exceptional performance, with metrics indicating near-perfect predictive capabilities. The model achieves an accuracy of 0.99788, precision of 0.997879, recall of 0.997885, F1-Score of 0.997880, and a ROC-AUC score of 0.99996. The confusion matrix further supports these results, showing minimal misclassifications and indicating that the majority of instances are correctly classified. The AUC-ROC scores for each class—normal (1.00000), DOS (1.00000), PROBE (1.00000), R2L (1.00000), and U2R (0.99999)—highlight the model's excellent capability to distinguish between different attack types and normal traffic. The values are depicted in Figure 4.13

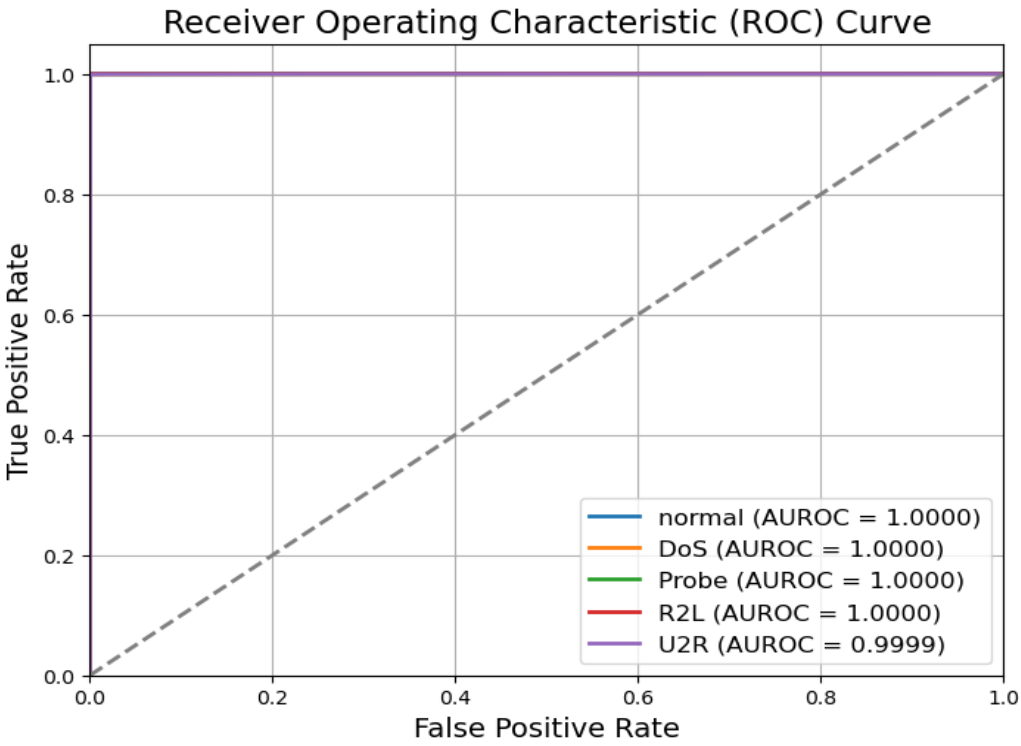


Figure 4.12: Area Under the ROC Curve of Soft Vote Classifier
Source: Researcher (2024)

This robustness in classification is critical for network intrusion detection, ensuring high precision and recall across all categories. The ensemble's ability to integrate the predictions from various models, balancing their strengths and weaknesses, leads to a superior overall performance, making it an ideal choice for this task. The strategic weighting of the models further enhances its effectiveness, providing a highly accurate and reliable solution for detecting network intrusions.

4.6 Discussion

In this section, the various models are analyzed to classify malware threats. Secondly, this research work is compared with two other research works in literature that used more than one dataset in training, analyzing and testing the models. These datasets are tagged as legacy dataset NSL-KDD dataset, UNSW-NB15 and ECML/PKDD2007 Dataset; while the modern datasets are CSE-CIC-IDS2018, CIC-IDS2017 and CSIC-

HTTP 2010. The models trained and analyzed include Random Forest (RF), Decision Tree (DT), K-Nearest Neighbor (KNN), Naïve Bayes (NB) and Logistic Regression (LR). The models were evaluated using the following cross-validation parameters Accuracy, Precision, Recall, F1-score, and Area under the ROC curve (AUC-ROC). Also used to plot the ROC curve are FPR and FNR values. Table 4.8 depicts the bench mark that contains analyzed values of the models trained and tested in this research work. The values contained in Table 4.9, Table 4.10, Table 4.11 and Table 4.12 are respectively compared and ranked with the bench mark values.

4.6.1 Comparative Analysis of the Models in this research work

The comparative analysis of different models reveals significant differences in their performance metrics as depicted in Table 4.8. The Random Forest Classifier outperforms other models with the highest accuracy of 0.99932 and an exceptional ROC-AUC score of 1.0, indicating its superior ability to distinguish between classes. The KNN model also shows strong performance, with an accuracy of 0.98816 and a ROC-AUC of 0.99740. In contrast, the Naive Bayes classifier demonstrates much lower performance across all metrics, with an accuracy of 0.39624 and a ROC-AUC of 0.7419, suggesting it struggles with the dataset's complexity. The Decision Tree Classifier performs nearly as well as the Random Forest with an accuracy of 0.99804 and a ROC-AUC of 0.9988, though its precision score appears to be incorrectly reported, matching that of the Naive Bayes Classifier. Overall, the Random Forest Classifier and Decision Tree Classifier exhibit the best performance, making them the preferred choices for this classification task.

Table 4.8: Comparative Analysis of the Models Used in this Research Work

Model	Accuracy	Precision	Recall	F1-Score	ROC
KNN Classifier	0.98816	0.988158	0.988170	0.988152	0.99740
Random Forest Classifier	0.99932	0.999322	0.988170	0.999319	1.0
Naive Bayes Classifier	0.39624	0.410736	0.396438	0.268998	0.7419
Decision Tree Classifier	0.99804	0.410736	0.998042	0.998041	0.9988
Logistic Regression	0.37945	0.26451	0.379460	0.259775	0.7421

Source: Researcher (2024)

4.6.2 Basis for comparing this research work with other works in literature

The works in literature compared with this research work operate under the same Android operating system (and can also use Windows operating system), used either ensemble learning approach or hybridization technique or both. The dataset used to train and analyze the models is NSL-KDD dataset. It is the desire of this researcher to compare, not only the performances of the trained models, but the contribution of the dataset used in the analysis to the performance of the models. Therefore, five other datasets are used in the comparison, to determine their importance or otherwise relative to NSL-KDD dataset. They include ECML/PKDD 2007 dataset, CSIC-HTTP 2010 dataset, CSE-CIC-IDS 2018 dataset, CIC-IDS 2017 dataset and UNSW-NB 15 dataset. These datasets are legacy datasets and more modern datasets, and they are applied by Chakir *et al.* (2023) and Saini *et al.* (2023) respectively. The base models trained include KNN, DT, NB and LR (either as single classifiers or in ensemble learning mode). They have a common intent of detecting malware or web-based attacks on host

systems and the cloud using permissions, system call logs, and API call logs. The analysis was done using cloud computing technology, ensemble learning and hybridization of ensemble learning techniques. However, while this work uses Random Forest (an ensemble bagging technique) to train the base models (DT, NB and LR), the other works hybridized RF and XGBoost predictions in addition to determine the strength of the resulting common classifier. KNN dataset does not train its models per se, rather it uses Euclidean distance measure to determine the nearest neighbor of the k set and assigns the object to the majority class in the k set of nearest neighbors. A brief introduction of the datasets follows:

CSE-CIC-IDS 2018 Dataset

This dataset is a combination of the communications security establishment (CSE) and the Canadian Institute of Cyber-security (CIC). The two bodies jointly produced the dataset. It has 16.23 million records. However, it is an unbalanced dataset with 17% classes belonging to the abnormal attack types. It also has the potential of identifying new attack types, it is modern, real-world and practical in modern day use.

CIC-IDS2017 Dataset

This dataset comprises of benign and recent attack types, and closely reflect real-world data such as PCAP files. It uses CSV files among others. Attack types in the dataset include DDOS, Botnets, Infiltration, web attacks, Heart bleed, brute force, DOS, Brute force SSH and FTP, and more.

UNSW-NB15

This dataset was created by the University of New South Wales (UNSW). Its raw network packets were generated using IXIA (a modern network testing program)

perfect storm program in UNSW. It has up to nine attack types with 49 features and a class label.

NSL-KDD Dataset

This dataset is an improvement of KDDCup'99 dataset, however, it may not be a perfect real-world representation, but an adequate benchmark dataset to help researchers compare different intrusion detection methods. It includes different attack types, which are grouped into DOS, PROBE, R2L and U2R attack types and the Normal applications. It has 42 features, including the class label.

ECML/PKDD 2007 Dataset

This dataset was generated for ECML/PKDD (European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases) discovery challenge. It contains 24504 and 10,502 valid requests for training and testing respectively, and 15110 malicious requests for testing. It includes web-based attacks such as XSS (Cross-Site Scripting), SQLI (Structured Query Language Injection), LDAP (Lightweight Directory Access Protocol) injection.

CSIC HTTP 2010 Dataset

This dataset contains requests generated to an e-commerce application. It contains two types of requests: 36,000 normal requests for training, and 36,000 normal requests for testing, and 25,000 requests classified as attacks for testing. It contains many types of web attacks like SQLI, XSS, CRLF (Carriage return, ASCII 13, \r) Line feed, ASCII 10, \n) and buffer-overflow.

The problem with the two types of datasets is that they contain only valid training requests. Therefore, they do not provide a good model that trains only with normal data

samples. As such, they will not be able to detect attacks that mimic normal activity. Hence, the authors extracted samples from attack dataset to mix with the normal dataset to use for intrusion detection.

4.6.3 Comparison of Results in this research work with That of Other Works in Literature

The comparison is done according to the datasets, each of these datasets is used to train and analyze the models.

NSL-KDD Dataset

This dataset was used to train and analyze models in Saini *et al.* (2023) and the results are depicted in Table 4.9.

Table 4.9: Performance Metrics for NSL-KDD DATASET

	Models	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	TNR (%)	FPR (%)	FNR (%)
1	KNN	95.03	94.14	95.19	94.67	94.89	5.11	4.81
2	DT	98.85	98.67	98.85	98.76	98.85	1.15	1.15
3	RF	99.15	99.14	99.03	99.08	99.26	0.74	0.97
4	NB	85.66	90.52	77.13	83.29	93.02	6.98	22.87
5	LR	87.39	88.92	83.14	85.93	96.06	0.94	16.86
6	HV	99.24	99.28	99.09	99.18	99.38	0.62	0.91

Source: Saini *et al.*, (2023)

Where:

KNN – K-Nearest Neighbor Classifier.
DT – Decision Tree Classifier
RF – Random Forest Classifier
NB - Naïve Bayes Classifier
LR - Logistic Regression Classifier
HV - Hard Vote Classifier

From Table 4.9, the accuracy of this research work, Table 4.8, outperformed that of Table 4.9 with RF (99.93%) as against HV (99.24%). Similarly, the ensemble classifier value of this research work, Table 4.8, SV (99.79%) outperformed the hard vote value of Table 4.9 HV (99.24%). However, the least value of Table 4.9 NB (85.66%) outperformed that of this research work, which is LR (37.94%) in Table 4.8.

Similarly, the highest F1-Score RF (99.93%) of Table 4.8 outperformed that of Table 4.9 HV (99.18%), but the lowest value of F1-Score in Table 4.9, NB (83.29%) outperformed that of Table 4.8 NB (26.89%). Again, the precision value of Table 4.8 in this research work, RF (99.93%) outperformed that is Table 4.9 HV (99.23%). However, the lowest precision value of Table 4.9 LR (88.92%) outperformed that of Table 4.8, which are two with the same value NB (41.07) and DT (41.07%).

The AUC-BOC of this research work's dataset RF (1.00) outperformed that of Table 4.9, which is HV (0.9923). This value indicates that RF is the best model in this research work and more preferable to hard vote of NSL-KDD dataset in Saini *et al.* (2023).

CSE-CIC-IDS2018 Dataset

It is pertinent to observe that this dataset is one of the modern datasets, and it is used to compare with NSL-KDD dataset used in this research work, considered a legacy dataset. Table 4.10 depicts the results of this dataset, which will be compared with the bench mark values in Table 4.8. The highest accuracy value of this research work in Table 4.8 is RF (99.93%). It outperformed that of Table 4.10, which is HV (98.92%). Similarly, the soft vote ensemble value of the research work Table 4.8 is SV (99.78%) and outperforms the ensemble value of Table 4.10, which is HV (98.92%).

	Models	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	TNR (%)	FPR (%)	FNR (%)
1	KNN	98.40	98.56	98.22	98.39	98.58	1.42	1.78
2	DT	98.31	98.56	98.04	98.30	98.58	1.42	1.96
3	RF	98.66	99.58	97.72	98.64	99.60	0.40	2.28
4	NB	86.77	81.45	95.08	87.74	78.53	21.47	4.92
5	LR	85.04	82.48	88.83	85.54	81.29	18.71	11.17
6	HV	98.92	99.47	98.35	98.90	99.48	0.52	1.65

Table 4.10: Performance Metrics for CSE-CIC-IDS2018 Dataset

Source: Saini *et al.* (2023)

The F1-Score value of the bench mark table (Table 4.8) is RF (99.93%), it outperformed the highest value of F1-Score in Table 4.10, which is HV (98.90%). However, the lowest F1-Score value of Table 4.10 LR (85.54%) outperformed that of Table 4.8, which NB (26.89). The precision value of this research work RF (99.93%) outperformed that of the Table 4.10, which is RF (99.54%). Indeed, the least precision value of Table 4.10 is NB (81.45%) and outperformed that of this research work, which is NB (41.07%) and DT (41.07%).

The highest value of AUC-ROC of the bench mark dataset NSL-KDD dataset is RF (1.00), and that of CSE-CIC-IDS2018 ensemble classifier is HV (98.913). Again, the research work dataset outperformed the modern dataset. It is pertinent to point out here that although NSL-KDD dataset is legacy dataset, it is a well composed dataset with normal and attack records, whereas the modern dataset is grossly imbalanced with only normal dataset. And attack features were only extracted from another evil dataset and fused into the dataset tor it to be used for this detection exercise (Saini *et al.* 2023).

CIS-IDS2017 Dataset

This modern dataset also primarily contains normal training and training datasets, but the researchers of the paper had to extract and infuse attack APT (Advanced Persistent Threats) into the dataset to detect abnormal threats in the dataset (Saini *et al.* 2023). Comparing the accuracy in Table 4.11, it is observed that the value of accuracy in Table 4.8, the bench mark values, outperformed that of Table 4.11. The obtained values are RF (99.93%) and HV (99.91%) respectively.

Table 4.11: Performance Metrics for CIC-DIS 2017 Dataset

	Models	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	TNR (%)	FPR (%)	FNR (%)
1	KNN	99.54	99.27	99.82	99.54	99.26	0.74	0.19
2	DT	99.79	99.77	99.81	99.79	99.77	0.23	0.19
3	RF	99.88	99.92	99.84	99.88	99.92	0.08	0.16
4	NB	85.06	79.07	95.54	86.53	74.49	25.51	4.46
5	LR	93.04	91.29	95.23	93.22	90.83	9.17	4.77
6	HV	99.91	99.88	99.95	99.91	99.88	0.12	0.05

Source: Saini *et al.* (2023)

However, the least model performance in the said dataset NB (85.06%) outperformed that of the benchmark dataset, which is LR (37.94%). Similarly, F1-score highest value of RF (99.93%) in Table 4.8 outperformed that of Table 4.11, which is HV (99.91%). However, the least performed model in terms of F1-score parameter in Table 4.11 with NB (86.53%) outperformed that of Table 4.8, which is LR (26.89%).

The precision values are very keenly contested, as that of Table 4.8 outperformed that of Table 4.11 with RF (99.93%) as against HV (99.92%), but the lowest model performance of Table 4.11, which is NB (79.07%) outperformed that if Table 4.8, which is NB (41.07%) and DT (41.07%) respectively.

UNSW-NB15 Dataset

This dataset is also regarded as a legacy dataset. The essence of comparison is because it used the same training and testing features as NSL-KDD dataset of this research work. Referring to Table 4.12, the accuracy highest value of RF (99.93%) of Table 4.8 outperforms that of the Table 4.12, which is HV (97.11%). And the lowest model performance of Table 4.12, which is NB (95.04%) indeed outperformed the lowest model value of Table 4.8 (the bench mark), which is LR (37.945%). The F1-Score value of this work, RF (99.93%) also outperformed that of the competing dataset, Table 4.12, which is LR (99.66%). However, the lowest performing model of F1-Score value in Table 4.12 is KNN (95.53%) and it outperformed that of the benchmark table (Table 4.8), which is NB (26.89%).

	Models	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	TNR (%)	FPR (%)	FNR (%)
1	KNN	95.50	96.30	95.56	95.53	95.43	4.57	4.44

2	DT	96.24	96.72	96.49	96.60	95.92	4.04	3.51
3	RF	96.98	96.32	98.31	97.31	95.33	4.67	1.69
4	NB	95.04	93.20	98.22	95.65	91.07	8.93	1.78
5	LR	96.18	93.84	99.66	99.66	91.85	8.15	0.34
6	HV	97.11	95.89	99.04	97.44	94.71	5.29	0.96

Table 4.12: Performance Metrics for UNSW-NB15 Dataset

Source: Saini *et al.*, (2023)

Similarly, the highest precision score of Table 4.8, which is RF (99.93%) outperformed that of Table 4.12, which is DT (96.72%); and the lowest performing model in Table 4.12, which is NB (93.20%) outperformed that of Table 4.8, which is NB (41.07%) and DT (41.07%) respectively.

ECML/PKDD2007 Dataset

This dataset was used by Chakir *et al.* (2023) with the same training algorithm and base models as the dataset used by this research work, NSL-KDD dataset. The comparison is based on the information extracted and depicted in Table 4.13. The highest accuracy value of this research work, which is depicted in Table 4.8 with RF (99.93%) outperformed the highest accuracy value in Table 4.13, which is RF (99.597%). However, the least accuracy value in this dataset (ECML/PKDD2007 Dataset), which is KNN (95.11%) outperforms that of this research work LR (37.945%). Similarly, the highest F1-Score value of this research work, which is RF (99.93%) outperformed that of the contending paper in Table 4.13, which is RF (99.129%), and the lowest model performance of Table 4.13 is KNN (88.112%) outperforms the least value in this research work, which NB (26.89%), Table 4.8 refers. Again, the highest precision value of Table 4.13, which is NB (99.952%) outperformed that of this research work, which is RF (99.93%), even its lowest model performance value that is DT (96.383%) still

outperforms that of this research work, which is NB (41.07%) and DT (41.07%) respectively. The highest value of AUC-ROC of this paper is equal to 1.0 from the HV and RF respectively. Although this research work also recorded RF (1.00) AUC, the soft vote value is indeed lower than that of Table4.13 with a value of SV (0.99960). With the high values of AUC recorded in this dataset by two models, HV and RF, which gives it an edge over this research work, the high accuracy value notwithstanding, it is the opinion of this researcher that this dataset performed creditably well with a better overall performance as against this research work. From literature, the higher the AUC value, the better the model performance. More so, ECML/PKDD2007 Dataset has a higher precision value compared to that of this research work, and, the higher the precision value, the better the model performance. These values led the researcher to conclude that this dataset has an edge over NSL-KDD dataset.

Table 4.13; Performance Evaluation for Each Classifier Based on the Dataset.

ECML/PKDD 2007 Dataset						
Classifier	A (%)	R (%)	P (%)	F1 (%)	FPR (%)	FNR (%)
NB	96.07	82.914	99.952	90.639	0.912	17.086
KNN	95.112	79.002	99.597	88.112	0.095	20.998
DT	99.140	100	96.583	98.158	1.116	0
LR	97.776	92.335	97.843	95.009	0.606	7.665
RF	99.597	100	98.274	99.129	0.523	0
HV	99.451	99.880	97.773	98.815	0.677	0.120

Source: Chakir *et al.* (2023)

Where

NB: Naïve Bayes Classifier
KNN: K-Nearest Neighbor Classifier
DT: Decision Tree Classifier
LR: Logistic Regression Classifier
RF: Random Forest Algorithm
HV: Hard Vote Classifier
A: Accuracy

R: Recall
P: Precision
F1: F1-Score
FPR: False Positive Rate
FNR: False Negative Rate

CIC- HTTP 2010 Dataset

This dataset is considered a relatively modern dataset with more of normal training and testing dataset. The dataset has imbalanced features that needed the use of SMOTE, Pearson Correlation and Information gain (IG) to balance the data labels before preprocessing. The dataset is then split to train and test the model in a ratio of 80:20, and after training, prediction of the test dataset is done and the models' performances are evaluated on the dataset. The performance parameters are contained in Table 4.14.

Table 4.14; Performance Evaluation for Each Classifier Based on Dataset

CSIC HTTP 2010 Dataset						
Classifier	A (%)	R (%)	P (%)	F1 (%)	FPR (%)	FNR (%)
NB	93.100	93.100	93.181	93.097	4.733	9.067
KNN	88.533	88.633	89.257	88.588	5.067	17.667
DT	99.140	100	96.383	98.158	1.116	0
LR	97.776	92.335	97.843	95.009	0.606	7.665
RF	99.867	99.867	99.867	99.867	0.267	0
HV	98.367	98.367	98.378	98.367	2.400	0.867

Source: Chakir *et al.* (2023)

From Table 4.14, it is observed that the highest accuracy value is RF (99.867%), it is outperformed by the highest accuracy of this research work RF (99.93%), however, the lowest accuracy value of the competing dataset, which is KNN (88.633%), outperformed that of this research work, which is LR (37.94%). Again, the F1-Score

value of this research work, RF (99.93%), outperformed that of Table 4.14, which is RF (99.867%), but its lowest model performance, which is KNN (88.588%) also outperformed that of this research work, in Table 4.8, which is NB (26.89%).

Similarly, the precision value of this research work, RF (99.93%) outperformed that of the Table 4.14, which is RF (99.867%), but its lowest model performance in terms of precision, KNN (89.257%) out smarted that of this research work, which is NB (41.07%) and DT (41.07%) respectively. It is also observed that AUC-ROC of this research work, RF (1.00) outperformed that of the work in literature, as depicted in Table 4.14, which is HV (0.96875).

Since high AUC implies that the model is better, then RF is a better model as against the HV model of the work's ensemble learning problem.

4.6.4 Lessons drawn from comparing this work with works in literature

i. The principle of producing better detection results by combining (hybridization or ensemble learning) classifiers against single classifiers, may not always be true. Depending on the problem at hand, single classifiers may do better than ensemble hard or soft vote classifiers, as evident in these works, and vice versa (Saini *et al.* 2023).

ii. The tradition of comparing the efficacy of trained models using a single dataset, may not always produce better results as against the use of multiple datasets on a single set of training models (Domingos, 2012; Saini *et al.* 2023; Chakir *et al.* 2023).

iii. The most known and widely used datasets were generated for network-based detection and may not be suitable for web-based attack detection solutions because they do not contain real-world attack samples. This has made

researchers to seek new and modern datasets for use in research works. However, this has not proved very effective as can be seen in the results of this comparison between modern and legacy attack sets, giving legacy attack sets leverage over the modern ones that mostly contain normal data features (Saini *et al.* 2023; Chakir *et al.* 2023).

iv. Traditionally, datasets are mostly imbalanced, with normal and DOS attack types having more records than some other attack types, which usually make accuracy of the model performance not always reliable, emphasis is rather shifted to precision and F1-score. Even the AUC results are best when the dataset is imbalanced (Saini *et al.* 2023). Evidence has shown that with the use of SMOTE, Pearson correlation, and information gain (IG) to balance and preprocess the dataset before normalization and reduction of features using PCA has yielded very good results, irrespective of the age of the dataset.

4.7. Publications

4.8 Ethical issues

4.8.1 Conflicts of Interest

This research work does not have any conflict of interest or personal relationship with a third party whose interest can be positively or negatively influenced by the content of this research work.

4.8.2 Citation

NSL-KDD data set, obtained from Kaggle repository, is open source, and it is used purely for research purpose and the source duly acknowledged. No alteration to the content of the dataset was made. Sources used in the body of this work are appropriately acknowledged and cited.

CHAPTER FIVE
SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Preamble

This chapter gives an overview of this research work. It summarizes the importance of electronic healthcare system (EHS) in nation building, for health is wealth; its security implications, especially to the patient, with particular emphasis on ethics in handling patients' records. EHS system is constantly targeted by malware and cyber attackers seeking to steal and extract sensitive information, and is also subjected to tactics aimed at intimidating, harassing, and disrupting the system as well as the patients it serves. The stolen information is sold for money. Efforts are made to detect the intrusions by combining algorithms to produce a more formidable model that will produce better detection through the use of voting classification. The dataset used to train the models is NSL-KDD dataset. The trained models predict the labels, which are used to compare with the expected labels using confusion matrix; and their performances are measured using cross validation parameters such as accuracy, precision, recall, F1-score and AUC. Also, contribution to knowledge and benefits of this research work to the hospital, patients and the society at large are pointed out.

5.2 Summary

Electronic healthcare system (EHS) has seriously outperformed traditional paper-based healthcare system, which unfortunately is still being used in Nigerian hospitals. With EHS, hospital personnel can assess patient's records with ease, share information between departments, units and even hospitals for inter-operability.

These benefits endeared hospitals, patients and even the staff, as they reduce the risk of patients' privacy abuse by internal and external intruders. Some malware developers'

intent is to make money from the innocuous user by distorting communication, stealing important information and even blackmail through espionage and ransomware.

This research work attempts to salvage the security implications of using mobile devices, laptops and PCs and other network-based equipment and the cloud to foster healthcare in hospitals. It combines machine learning (ML) algorithms, through ensemble learning method (bagging), using Random Forest algorithm to train base models (DT, NB and LR); and KNN. The predictions of these classifiers were aggregated to a formidable mega model (soft vote classifier) that effectively detects malware threats. The proposed system monitors applications requesting for permissions to either install, update applications, or even mischievously subvert operations, from the user, who could be the doctor, nurse, lab technician, pharmacist or even the patient. Such dangerous requests are detected and reported by intrusion detection system (IDS) to the system Administrator, the user or intrusion response system (IRS) for necessary actions.

The proposed system is developed using python programming language, in conjunction with its external libraries such as scikit-learn, pandas and NumPy, to mention but a few. It is evaluated for performance using confusion matrix standard metrics such as accuracy, precision, recall, and f-measure. This research work attained 99.93% (RF) accuracy, precision, and F1-score respectively; and an overall AUC score of 1.00 (RF). Indeed, the RF value outperformed the ensemble soft vote value of 99.79% (SV), and has been confirmed by the AUC-ROC value to be the best choice classifier of malware threats. Individual models scored and accuracy of 98.82% (KNN), 39.62% (NB), 99.80% (DT) and 37.94% (LR). Similarly, the models had F1-Score of 98.81% (KNN), 26.89% (NB), 99.80% (DT), and 25.97% (LR). The results evidenced that LR struggled with the complexity of the dataset in analyzing and detecting malware threats.

The attack labels that resulted from these analyses include Normal, DOS, PROBE, R2L and U2R. The predictions of this research work were compared with that of two other works in literature piloted by Saini *et al.* (2023) and Chakir *et al.* (2023). These papers introduced the use of multiple datasets in training and testing a specific set of models. Although the results from these works came from both legacy and modern datasets, they could not perform as good as NSL-KDD dataset used by this research work, which was properly balanced using SMOTE, Pearson correlation and information gain (IG) data extraction, with normalization and feature selection using PCA. Feature imbalance in ML datasets has been a problem in carrying out effective malware detection by trained models without bias and over fitting.

5.3 Conclusion

The results of this study indicate that ensemble learning methods do not always outperform single classifiers in network-based attack detection, as it is commonly presented in literature (Chakir *et al., 2023*). Both have their strengths and weaknesses. Ensemble learning approaches exhibit higher accuracy, precision and low false positive rate compared to single classifiers. However, in terms of recall, FNR, training and prediction time, single classifiers perform better.

The results presented in this work have important implications in efforts made by research community to combat the menace of malware threats, by emphasizing the importance of electronic healthcare system in preference to the traditional paper-based healthcare system. Though EHS is not without challenges, it is the most preferred hospital management system globally. This research work has successfully combined machine learning tools (Random Forest algorithm, NB, KNN, DT and LR models) to

improve malware threat detection and reporting using cloud computing technology (CCT) in healthcare systems, and especially hospitals in Nigeria.

Secondly, using more classifiers (ensemble learning) in an effort to curb the menace of malware has paid-off, compared to the use of single classifiers.

Thirdly, EHS has enabled the sharing of patients' records between and within healthcare professionals, for effective healthcare delivery, within ethical principles. The safety and respect of patients' privacy has been enshrined in this research work, by ensuring that only authorized persons and applications have access to patients' records.

One of the common problems in machine learning processes is the database, especially public databases that are modern, recent and contain practical normal and malicious applications for use by researchers to conduct malware detection and control. The current databases commonly used are the KDD family and UNSW-NB15, which are no longer suitable particularly in web-based attack detection. However, the most recent databases include ECS-CIC-IDS2018 and CIC-IDS2017, which have their shortcomings. For instance, they have only 2180 and 928 web based attack instances. In particular, they contain only normal applications, and not fit for training and testing malware presence in the cloud. The challenge then is to create a novel all rounded database that will contain both normal and malware applications with modern and recent real-world problems.

The dataset used in this research work is deemed legacy. But the use of SMOTE, Pearson correlation and information gain to balance the data distribution and extraction, placed it over and above the modern ones. Proper feature selection and extraction is critical in model training for better performance.

5.4 Scientific Implications of Findings

i. Data generated these days by hospital management are huge and unstructured. They are generated in the cloud, stored in the cloud (use of servers with huge disk spaces), processed in the cloud, and results transmitted to the device through its IP address as client. Thus, reducing the risk of pests, occupation of huge storage spaces, difficulty of searching for patient's information and the time, and usurping the overhead of physical security. The secured internet, policed by this product, will enable researches with good results, as malware may not even find space to avert controlled environment (sand boxes).

ii. The application of this research work will assist patients in assessing their health records, be it in the hospital or at home. Thus, making healthcare delivery patient- centered.

iii. The introduction of intelligent techniques to detect malware threats in electronic healthcare system is timely, especially these days that COVID'19 is ravaging the world with many variants, and now subvariants of omicron are circulating. The use of this product will enable the sharing of medical records of people traveling into the country, and vice versa. Thus, reducing the embarrassing situation of quarantining every traveler that enters the country from other countries, and reduce cost.

5.5 Contribution to Knowledge

i. Pham *et al.,* (2018) trained tree classifiers such as decision tree (DT) and random forest (RF) using Adaboost algorithm and NSL-KDD dataset, the result obtained was 80.59% (DT) and 80.07% (RF). This research work improved on their work by training more classifiers such as Random Forest

(RF), K-nearest neighbor (KNN), decision tree (DT), logistic regression (LR) and Naïve Bayes (NB) using NSL-KDD dataset and got the following accuracies: 99.93% (RF), 99.78% (SV), 99.80% (DT), 98.82% (KNN) and 37.94% (LR) and 39.62% (NB).

ii. Pattern matching of target object with signatures of malware families in databases had been the old process, however, this research work has trained classifiers to learn the behavior and characteristics of 67,343 normal applications and use that knowledge to detect malware, devoid of signatures of malware families, as malware developers have outsmarted the process using obfuscation techniques.

iii. Saini *et al.* (2023) introduced the use of multiple datasets in the search for effective method to detect malware threats. Their results were outperformed by this research work with just a single dataset, which has balanced features, using SMOTE.

iv. To the best of our knowledge, the traditional believe that combining ML classifiers or models in detecting malware threats would perform better than single classifiers, has been proved not to be true in all cases. Depending on the type of problem being solved, single classifiers have performed better than ensemble learning consensus classifier in some cases. For instance, RF outperformed soft vote classifier with 99.93% accuracy as against 99.78% respectively.

5.6 Suggestions for Future works

i. **Implementation** - To implement these developed and tested techniques, it is recommended that they be integrated to Application Programming Interface (API), and uploaded to the cloud. Then the front end (user interface

- UI) should be developed using JavaScript, HTML and software code control (SCC) system. The UI will interact with the model at the back end via the API when an application is input through it for analysis.

ii. **Class imbalance** – Malware datasets are often skewed, which can lead to bias and affect the performance of trained models. There is, therefore, the need to look into the imbalanced issue to ensure accurate detection across all malware categories. For instance, most malware are greatly outnumbered by normal type and DOS attack type as against PROBE, R2L and U2R attack types.

iii. **Single execution Path** – Dynamic analysis method analyses and detects malware threats from a single execution path. Multiple execution paths should be explored to enable the system to be able to identify different behaviors displayed by the suspicious executable files.

iv. Resource sharing - There is still difficulty associated with sharing resources in the cloud, especially patients' records and other security issues. This should be investigated.

REFERENCES

Abraham, S. (2017). *List of Types of Malware*, MalwareFox. https://www.malwarefox.com/malware-types/ (Retrieved on 12th June, 2021)

Abushark, Y. B., Khan, A. I., Alsohami, F., Almalawi, A., Alam, M.M., Agrawal, A., Kumar, R. and Khan, R. A. (2022). Cyber Security Analysis and Evaluation for Intrusion Detection Systems. *Computers, Materials and Continua.* DOI: 10.32604/cmc.2022.025604

Aijaz, M., Nazir, M. and Mohammad, M.N. (2023). Thread Modeling and Assessment Methods in the Health Care IT System; A Critical Review and Systematic Evaluation. *Springer Nature Computer science,* 4:714. https://doi.org/10.1007/s42979-023-02221-1

Akram, F., Liu, D., Zhao, P., Kryvinska, N., Abbas, S., and Rizwan, M. (2021).. Trusworthy Intrusion Detection in E-Healthcare Systems. *Frontiers in Public Health, 9:788347.* DOI: 10.3389/fpubh.2021.788349

Al-Ajlan, A. (2015). The Comparison Between Forward and Backward Chaining. *Internal Journal of Machine Learning and Computing*, vol. 5, No. 2. DOI: 10.7763/IJMLC.2015.v5.492.

Alder, S. (2024). Security Breaches in Health Care in 2023. *The HIPAA Journal*

Aljawarneh, S., Aldwairi, M., and Yassrin, M.B. (2017). Anomaly-Based Intrusion Detection System Through Feature Selection Analysis and Building Hybrid Efficient Model. ELSEVIER. *Journal of Computer Science*. https://dx.doi.org/10.1016/j.jocs.2017.03.006.

Amazon Web Services (AWS) (2021). *Amazon Machine Learning: Developer Guide*. Amazon Web Services Inc. http://aws.amazon.com.

Anthony, R. A. (2014). A Study on Data Mining Based Intrusion Detection System. *International Journal of Innovative Research in Advanced Engineering* (IJIRAE),3 (1),.

Anton, H. and Rorres, C. (2014). *Elementary Linear Algebra*, Applications Version, 11-th Edition. ISBN 978-43441-3. Wileyplus.com.

Arshad, S; Khan, A; Shah, M.A., and Ahmed, M. (2016). Android Malware Detection and Protection: A Survey. *International Journal of Advanced Computer Science and Applications*. (IJACSA), 7(2).

Atkinson, M. (2015). *An Analysis of Android Application Permissions*. Pew Research Center, Internet and Technology.

Attah, A. O. (2017).Implementing the Electronic Health Record in Nigeria: Prospects and Challenges. A Master's Thesis in Telemedicine and E-Health (TLM-3902).The Arctic University of Norway.

Azad, C. and Jha, U. K. (2014). Data Mining Based Hybrid Intrusion Detection Systems. *International Journal of Science and Technology*, vol 7(6): 781 - 789,

Babita, E. and Kaur, G. (2017). A Review: Network Security Based on Cryptography and Steganography Techniques. *International Journal of Advanced Research in Computer Science.* Vol. 8, No.4. E-ISSN: 0976-5697

Baby, J.J. and Jeba, J. R. (2017). Survey Paper on Various Hybrid and Anomaly Based Network Intrusion Detection Systems. *Research Journal of Applied Sciences.* 2(3-4): 304 – 310.

Berrar, D. (2018). Cross Validation. *Encyclopadia of Bioinformatics and Computational Biology,* vol. 1, ELSEVIER, pp. 542 – 545. https://dio.org/10.1016/B978-0-12-809633-820349-x

Bhandari, A. (2020). Every Thing you Should Know About Confusion Matrix for Machine Learning. *Analytics, Vadhya, 17.*

Bhuyan, M. H., Bhattacharyya, D. K., and Kalita, J. K. (2014). *Network Anomaly Detection: Methods, Systems and Tools*. IEEE Communications Surveys and Tutorials, 16(1).

Breiman, L. (2001) Random forest. *Machine learning,* 45 (1): 5-32

Brownlee, J. (2021). How to Combine Predictions for Ensemble Learning. *Machine Learning Mastery* (E-BOOK).

Bui, L.T., Vu, V.T., and Dinhm T.T. H. (2017). A Novel Evolutionary Multi-Objective Ensemble Learning Approach for Forecasting Currency Exchange Rates. *Data knowledge Engineering*. https://dx.doi.org/10.1016/j.datak.2017.07.001

Cai, L., Li, Y., Xiong, Z. (2020). JOWMDroid: Android Malware Detection Based on Feature Weighting with Joint Optimization of Weight-Mapping and classifier Parameters. *Computer Society, 102086. https://doi.org/10.1016/j.cose.2020.102086*

Chakir, O., Rehaimi, A., Sadqi, Y., Alaoui, E. A. A., Cridun, M., Gaba, G. S., and Gurtov, A. (2023). An Empirical Assessment of Ensemble Methods and Traditional Machine Learning Techniques for Web-based Attack Detection in Industry 5.0. *Journal of King Saud University - Computer and Information sciences,* 35(2023):103 – 119. https://dio.org/10.1018/j.jksuci.2023.02.009

Cheng, D., Zhang, S., Deng, Z., Zhu, Y and Zong, M. (2014). KNN Algorithm With Data-Driven K Value. *Springer International Publishing,* Switzerland. Pp. 499 – 512.

Cheng, L., Liu, F., Yao, D. (2017). *Enterprise Data Breach: Causes, Challenges, Prevention, and Future Directions.* WI8RES Data Mining Knowledge Discovery. DOI: 10.1002/widm.1211.

Chumachenko, K. (2017). *Machine Learning Methods for Malware Detection and Classification.* Bachelor's Thesis, Information Technology.

Comsast Business (2021). DDOS Threat Report: DDOS Becomes a Bigger Priority as Multi-Vector Attacks are on the Rise. (Retrieved May 20, 2024). https://corporate.comsast.com/press/releases.

Dewa, Z. and Maglaras, L. A. (2016). Data Mining and Intrusion Detection Systems. *International Journal of Advanced Computer Science and Applications.* 30 (30).

Dewa, Z. and Maglaras, L. A. (2016). Data Mining and Intrusion Detection Systems. *International Journal of Advanced Computer Science and Applications.* 30 (30).

Domingos, P. (2012). A Few Useful Things to Know About Machine Learning. *Communication of the ACM*, 55(10).

Dong, D., Ye, Z., Su, J., Xie, S., Cao, Y., Kochan, R. (2020). *A Malware Detection Method Based on Improved Fireworks Algorithms and Support Vector Machine*, 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronic Telecommunications and Computer Engineering (TCSET). DOI: 10.1109/TCSET49122.2020.235556

Eysenbach, G. (2006). What is E-Health: *Journal of Medical internet Research, 3(2):e20.* DOI: 10.2196/jmir.3.2.e20

Fan, M., Ezeudoka, B.C. and Qalati, S. A. (2020). Exploring the Resistant to E-Health Services in Nigeeria: An Integrative Model Based pon the Theory of Planned Behavior and Stimulus-Organism.Response. *Humanities and Social Sciences Communication.* https://doi.org/10.1057/s41599-024-03090-6

Feizollah, A.., Anuar, N. B.., Salleh, R., Wmalina, F., Maarof, R. U. R. and Shamshirb, S. (2014). A Study of Machine Learning Classifiers for Anomaly-based Mobile Botnet Detection. *Malaysia Journal of Computer Science, 26(4):*251 – 256

Fix, E. and Hodges, J.L. (1951). Discriminatory Analysis. Non-parametric Discrimination; Consistency Properties. Technical Report 4, USAF School of Aviation Medicine, Randolph Field, TX, USA.

Freund, Y. and Schapire, R.E. (1996). *Experiments with new boosting algorithm.* Proceedings of the Thirteenth International Conference on Machine Learning. Pp. 148 – 156

Garg, S and Paliyan, N. (2019). A Novel Parallel Classifier Scheme for Vulnerability Detection in Android. *Computer Electronic Engineering, 77(2019):12 – 26.* *https://doi.org/10.1016/j.ccompdeceng.2019.04.019*

GitHub Inc. (2020). *NSL-KDD Dataset*. https://www.github.com

GoodworkLabs (2018). How to Choose the Right Machine Learning Algorithm, *Machine learning (Blog).* 7100 Stevenson Blvd, Fremont CA, 94538, US

Grandini, M., Ragli, E. and Visani, O. (2020). Metrics for Multi-class Classification: An Overview, arXiv Preprint arXiv: 2008.05756

GSMA (2019). *Mobild Telecommunications Security Threat Landscape*. Floor 2, The Walbrook Building, 25 Walbrook, London EC4N 8AF United Kingdom.

Hagan.T. Demuth, H.B., Beale, M. (1996). *Neural Network design,* PWS Publishing Co. Boston MA: USA.

Harlalka, R. (2018). *Choosing the Right Machine Learning Algorithm*. @RajatHarlalka. (Retrieved on 15th March, 2021)

Hathaliya, J.J. and Tanwar, S. (2020). An Exhaustive Survey on Security and Privacy Issues in Healthcare 4.0. *Computer Communications,153(2020):311 – 335*

Health and Human Services (HHS) (2022). Electronic Medical Records in Healthcare. *Leadership for IT Security and Privacy. Across HHS.*

Heidari, M (2017). In Weld Defect Detection on Digital X-radiography Images, University of Oklahoma.

Heinold, B. (2016). *A Practical Introduction to python Programming*, http://www.brainheinold.net/python/A_Practical_Introduction_to_Python_Programming_Heinold.pdf (Retrieved 15th February, 2018).

Hira, Z.M. and Gillies, D.F. (2015). A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data. *Advances in Bioinformatics*. http://dx.doi.org/10.1155/2015/198363

Holland, S. M. (2019). *Principal Component Analysis (PCA)*. Department of Geology, University of Georgia, Atbens; GA 30602-2501.

Ibrahim, J., Danlandi, T.A., and Aderinola, M, (2017). *Comparative analysis between wired and wireless technologies in communication*: A review. Proceedings of 99th The IIER International Conference, Mecca, Saudi Arabia.

ICCC FBI (2021). Internet Crime Report. (Retrieved May 20, 2024). https://www.ic3.gov/media/PHD/AnnualReport

Idris, I., Adeleke, I., Uduimoh, A.A. and Tom, J.J. (N.D). Design Framework of Cyber Security Solutions to Threats and Attacks on Critical Infrastructure of Electricity Power Systems of Nigeria Companies. *Elizade University, Ilara, Mukin, Nigeria.*

Ikram, S.T. and Cherukuri, A.K. (2016). Improving Accuracy of Inttrusion Detection Model Using PCA and Optimized SVM. *CIT Journal of Computing and Information Technology,* 24 (2):133 – 148.

Inayat, Z; Gani, A., Anuar, N. B; Khan, M. K., and Anwar, S. (2016). Intrusion Response System: Foundations, design, and Challenges. ELSEVIER *Journal of Network and Computer Applications*, 62):53 – 74.

Ishaku, T. (2011). Water Supply Dilemma in Nigerian Rural Communities: Looking Towards the sky for an Answer. *Journal of Water Resources and Protection, 03(08):593 – 606.* DOI: 10.4236/jwarp.2011.38069

Jenyo, I., Amusan, E. A. and Emuoyibo-farhe, J. O. (2023). A Trust Management System for the Nigerian Cyber-Health Community. *International Journal of Information Technology and Computer Science, 1: 9 – 20.* DOI: 10.5815/ijitcs.2023.01.02

Jiang, X., Mao, B., Guan, J., Huang, X. (2020). Android Malware Detection Using Fine-Grained Features. Hindawi *Scientific Programming,* vol. 2020. https://doi.org/10.1155/2020/51901.38

Jolliffe, I. T. and Cadima, J. (2016). *Principal component analysis: A review and recent developments*. Philosophical Transactions. http://dx.doi.org/10.1098/rsta/2015.0202

Katiyr, N, Tripathis, S., Mumar, P., Verman, S., Sahu, A. K. and Saxena, S. (2024). AI and Cyber security: Enhancing Threat Detection and Response with Machine Learning. *Education Administration, Theory and Practice*, 30(4):6273 – 6282.

Kohavi, R (1996). *Scaling up the accuracy of naïve-bayes classifiers: A decision-tree hybrid*. KDD'96: Proceedings of the second Internation Conference on Knowledge Discovery and Data Mining, pp. 202 – 207

Kotsiantis, S. B., Zaharakis, I, and Pintelas, P. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Emerging Artificial Intelligence Applications in Computer Engineering,* 160(1): 2- 24

Kruegel, C. (N.D). Full System Emulation. Achieving Successful Automated Dynamic Analysis of Evasive malware. Chris@hosline.com

Kulkarni, H and Hughes, J. T. (2022). Worker Equality in Renal Medicine. *Internal Medicine Journal,* 52(11): 1859 – 1862.

Kumar, D. A. and Das, S. K. (2023). Machine Learning Approach for malware Detection and Classirication Using Malware Analysis Framework. *International Journal of Intelligent Systems and Application in Engineering 9OKOSAE), 11(1): 330 – 335*

Kushala, M.V. AND Slaylaja, B. S. (2020). Recent Trends on Security Issues in Multi-Cliud Computing: A Survey. *In 2020 International Conference on Smart Electronics and Communications (ICOSEC), pp 777 -781, IEEE.*

Lee, T. and Mody, J. J. (2006) Behavioral Classification. *EICAR Conference, USA, 45(4): 1 – 17*

Lord, N. (2020). *Principle of Least Privilege,* Digital Guardian, Waltham, MA 02451. http://www.digitalguardian.com/author/nate-lord

Markoulidakis, I., Kopsiaftis, G., Rallis, I. and Georgoulas, I (2021). Multi-class Confusion Matrix Reduction Method and its Application on net Promoter Score Classification Problem. *In the 14th Pervasive Technologies Related to Assistive Environments Conference (pp. 412 – 419)*

Mary Stella J. and Kumar, S. (2020). Prediction and Comparison Using Adaboost and ML Algorithms with Austictic Children Dataset. *International Journal of Engineering, Research and Technology* (IJERT), 9 (7).

Mondal, A., Paul, S., Goswami, R. T. and Nath, S. (2020). Cloud Computing Security Issues and Challenges: A Review. *In IEEE, pp. 15*

Moshiri, E., Abdullah, A. B., Azlina, R., Mahmood, B. R., and Muda, Z. (2017). Malware Classification Framework for Dynamic Analysis Using Information Theory. *Indian Journal of Science and Technology,* 10(21):1 – 14 DOI: 10.17485/ijst/2017/v10i21/100023

Narudin, F. A., Feizollah, A., Anuar, N. B., and Gani, A. (2014) Evolution of machine learning. *Springer-verlag,* Berlin, Heidelberg.New York NY 10001, US: Infobase publishing, p. 27

Nasteski, V. (2017). *An Overview of Supervised Machine Learning Methods*. HORIZONS. B, 4:51–52. DOI: 10.20544/HORIZONS.B. 04.1.17. P05.

Nguyen, T., McDonald, J. T., and Glisson, W. B. (2017). *Exploitation and detection of a malicious mobile application.* Proceedings of the 50th Hawaii International Conference on System Sciences. http://hdl.handle.net/10125/41911.

Nykanen, P. (2017). Implementation and Evaluation of E-Health Ecosystems in T-sided-Markets. *E-Health Two Sided Markets. Elservia Inc.*

Okediran, O., Sijuade, A., wahab, W., and Oladimeji, A. (2022). A Framework for a Cloud-based Electronic Health Record System for Nigeria. *LAUTECH Journal of Engineering and Technology, 1692):128 – 136*

Pawar, M.V. and Anuradha, J. (2015). *Network security and types of attacks in network.* International Conference on Litelligent Computing, Communication and

Convergence (ICCC – 2015), vol.48, pp 503 – 506. DOI: 10.1016/j.procs. 2015.04.126.

Pham, N.T., Foo, E., Suriadi, S., Jeffrey, H and Lahza H.F. (2018). *Improving performance of intrusion detection systems using ensemble methods and features Selection.* In (ACSW) Australian Computer Science Week, Brisbane, OID Australia. https://dio.org/10.1145/3167918.3167951

Pooja, M. R. and Pushpalatha, M.P. (2019). A Comparative Performance Evaluation of Hybrid and Ensemble Machine Learning Models. *.Journal of Health & Medial Informatics,* vol.10, Issue 2. ISSN 2157 – 7420

Prasath, V.B.S., Alfeilat, H.A.A., Lasassmeh, O., and Hassanat, A.B.A. (2017). Distance and Similarity Measures Effect on the Performance of K-Nearest Neighboir Classifier – A Review. arXiv:1708.04321v1[cs.LG] Artificial Intelligence(cs AI). https://doi.org/10.48550/arXiv.1708.04321

Ravanshad, A. (2018). *How to Choose Machine Learning Algorithm.* https://medium.com/@aravanslad/how-to-choose-machine-learning-algorithm-9a2a448e0df (Retrieved on 15th February, 2020).

Raymond, A., Schubauer, J. and Madappa, D. (2020). Over-Privileged Permissions: Using Technology and Design to Create Legal Compliance. *Journal of Business and Technology,* Kelley School of Business Research Paper No. 2020-57, Available at SSRN:
https://ssrn.com/abstract=3546518 or http://dx.doi.org/10.2139/ssrn.3546518

Resnik, D. B. (2020). What si Ethics in Research and Why is it Important? *National Institute of Environmental Health Sciences, pp 22*

Saeed, V. A. and Asaad, R.R.(2022) Cyber Security, Threats, Vulnerability, Challenges with Proposed Solution. *Applied Computing Journal, 2(4):227 – 244..* https://doi.org/10.52098/acj.202260

Saini, N., Kasaragod, V. B., Prakasha, K., Das, A. K. (2023). A Hybrid Machine Learning Models for Detecting APT Attacks Based on Network Behavior Anomaly Detection. *Concurrency and Computation: Practice and Experience Published by John Wiley and Sons.* https://dio.org/10.1002/cpe.7865

Salvador-Meneses, J., Ruiz-Chavez, Z., and Gracia-Rodriquez, J. (2019). Compressed KNN: K-Nearest Neighbor with Data Compression. Entropy. 21(2):34. DOI: 10.3390/e21030234

Santos, I., Devesa, J., Brezo, F., Nieves, J., Bringas, P. G. (2013). OPEM: A Static-Dynamic Approach for Machine Learning-Based Malware Detection. In: Harrero, A., Conference CISIS'12-ICEUTE'12-SOCO'12 Special Sessions. *Advances in Intelligent Systems and Computing, vo.189* Springer, Berlin, Heidelberg. https://dio.org/10.1007/987-3-642-33018-6-28

Saperito, G. (2019). A Deeper Dive into the NSL-KDD Dataset. *Towards Data Science.* (Retrieved May 28, 2023). https://towardsdtascience.com/a-deeper-dive-into-the-nsl_kdd-data-set-15c753364657

Sardi, A., Rizzi, A., Sorano, E. and Guerrien, A. (2020). Cyber Risk in Health Facilities: A Systematic Literature Review. *Sustainability,* 12, 7002.

Sharma, N., Oriaku, E.A., Oriaku, N. (2020). Cost and Effects of Data Breaches, Precautions and Disclosure Laws. *Intgernational Journal of Emerging Trends in Social Sciences,* vol. 8, Issue 1., pp. 33 – 41. DOI: 10.20448/2001.81.55.41

Shatnawi, A. S., Yassen, Q. and Yateem, A. (2022). An Android Malware Detection Approach Based on Static Feature Analysis Using Machine Learning Algorithms. *Procedia Computer Science, 201(c): 653 – 658. https://doi.org/10.1016/j.procs.2022.03.086*

Shi, H., Wang, W., Wu, P., Wang, D. (2020). *Support vector machine based on localized multiple kernel learning in pre-microRNA classification.* Proceedings of the 2nd International Conference on Electrical, Communication and Computer Engineering (ICECCE), Instabul, Turkey.

Singh, A and Chatterjee, K. (2019). Security and Privacy Issues of Electronic Healthcare System: A Survey. *Journal of Information and Optimization Sciences,* 40(8):1709 – 1729. DOI: 10.1080/02522667.2019.1703265

Singh, A. (2018). *A Comprehensive Guide to Ensemble Learning (with Python Code).* Analytics, Vidhya.(Retrieved on5th January, 2020)

Singh, A. K., Wadhwa, G., Ahuja, M., Soni, K., and Sharma, K (2020). Android Malware Detection Using LSI-based Reduced Opcode Feature Vector. *Procedia Computer Science, 173(2019)(2020):291 – 298.* https://doi.org/10.1016/j.procs.2020.06.034.

Singh, B. K., Verma, K., and Thoke, A. S. (2015). Investigation on Impact of Feature Normalization Techniques on Classifier's Performance in Breast Tumor Classification. *International Journal of Computer Applications* (0975 – 8887), 116 (19).

Singhal, A. and Cowie, M. (2020). What is E-Health? *E-Journal of cardiology Practices, 18(24):1 – 10*

Smmarwar, S. K., Gupta, G. P. and Kumar, S. (2024). Android Malware Detection and Identification Framework by Leveraging the Machine and Deep Learning Techniques: A Comprehensive Review. *Telematics and Informatica Report, 14(2024):100130*

Stein, J. (2020). *Data Breach Report,* Nortyh Carolina Department of Justice. www.ncdoj.gov/complaint

Sullivan, D.T. (2015). *Survey of Malware Threats and Recommendations to Improve Cyber Security for Industrial Control Systems* version 1.0. US Army Research Laboratory (ARL)

Sun, L., Li, Z., Yan, Q., Srisa-an, W., and Pan, Y. (2016). *SigPID: Significant Permission Identification for Android Malware Detection*. University of Nebraska, Lincoln, NE 68588.

Syed, A., Purushotham, K. and Shidagani, G. (2020). Cloud Storage Security Risks, Prctice and Meassures: A Review. *IEEE International Conference for Innovation in Technology (INOCON). https://doi.org.sdl.idm.oclc.org/10.1109/INOCON50539.2020.9298281*

Tatam, M., Shanmugam, B., Azam, S., and Kannoorpatti, S. A. K. (2021). A Review of Threat Modelling Approaches for APT Style Attacks. *Heliyon 7920210e05969.* https://doi.org/10.1016/j.heliyon.2021.e05909.

Vasiliadis, G., Polychronakis, M., Joannidis, S. (2014). "GPU-Assisted Malware", *International. Journal of. Information. Security*. vol. 14. No. 3. pp. 289 – 297.

Vemprala, N. and Dietrich, G. (2019). *A social network analysis (SNA) study on data breach concerns over social media*. Proceedings of the 52nd Hawaii International Conference on System Sciences. https://hdl.hanbdle.net/10125/60155.

Verizon (2018). 2018 *Data Breach Investigations Report*. www.verizonenterprise.com/Federal (Retrieved on 4 May, 2021).

Verizon (2022). Data Breach Investigations Report. (Retrieved May 20, 2024). https;//www.verizon.com/busness/resources/report

Wallisch, P. (2014). *MATLAB for Neuroscientists*. An Introduction to Scientific Computing in MATLAB. In MATLAB for Neuroscientists (Second Edition).

Wang, L., Gao, Y., Gao, S. and Yong, X. (2021). A New Feature Selection Method Based on a Self-variant Genetic Algorithm Applied to Android Malware Detection. *Symmetry, 13(17):1 – 21* https://doi.org/10.3390/sym.13071290

Wang, P. and Wang, Y. (2015). "Malware Behavioral Detection and Vaccine Development by Using a Support Vector Machine Model Classifier", *Journal of computer and system sciences.* 81:1012 – 1026

Wang, P. and Wang, Y. (2015). "Malware Behavioral Detection and Vaccine Development by Using a Support Vector Machine Model Classifier", *Journal of computer and system sciences.* 81: 1012 – 1026.

Wesolowski, T. E., Porwik, P. and Dorozi, R (2016). Electronic Health Record Security Based on Ensemble Classification of Keystroke Dynamics. *International Journal of Applied Artificial Intelligence,* 30(6):521 – 540. DOI: 10.1080/08839514.2016.1193715

Wu, W. C. and Hung, S. H. (2014). DroidDolphin: A Dynamic Android Malware Detection Framework Using Big Data and Machine Learning. *In: Proceedings of the 2014 Conference on Research in Adaptive and Convergent Systems, pp. 247 – 252.*

Yeng, P. K., Wulthusen, S. D., and Yang, B (2020) Comparative Analysis of Threat Modeling Methods for Cloud Computing Towards Healthcare Security Practices. *International Journal of Advanced Computer Science and Applications.* 11(11):772-784

Yeng, P. K.., Nweke, L. O., Yang,B., Fauzi, M. A., and Snekkenes, E. A. (2021). Artificial Intelligence-Based Framework for Analyzing Health Care Staff Security practice: Mapping, Review and Simulation Study. *JMIR Medical Informatics*, 9(120: e19250. https://medinform.jmir.org/2021/12/e19250

Yesilyurt, M and Yalman, Y. (2016). Security Threats on Mobile Devices and Their Effects; Estimation for the Future. *International Journal of Security and its Applications*. 10(2):13 – 26. http://dx.doi.org/10.14257/ljsla.2016.10.2.02

Yin, H. and Song, D. (2019). *Whole-System Fine-grained Taint Analysis for Automatic Malware Detection and Analysis*. https://www.researchgate.net/publication/2287857-54. (Retrieved on 10 March, 2021)

Yoti, J. and Saini, H. (2017). A Study on Networks and Comparison of Wired, Wireless and Optical Networks. *International Journal of Innovative Research in Computer and Communication Engineering,* 5(3)

Zhang, J. (2016). Introduction to Machine Learning: K-Nearest Neighbors. Annals of Translational Medicine, vol. 4(11):218. DOI: 10.21037/atm.2016.03.37.

APPENDIX I

NSL-KDD TRAINING DATASET

0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00,normal,20

0,udp,other,SF,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,1,0.00,0.00,0.00,0.00,0.08,0.15

,0.00,255,1,0.00,0.60,0.88,0.00,0.00,0.00,0.00,0.00,normal,15

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,123,6,1.00,1.00,0.00,0.00,0.05,0.07,

0.00,255,26,0.10,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19

0,tcp,http,SF,232,8153,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,5,0.20,0.20,0.00,0.00,1.00,0.0

0,0.00,30,255,1.00,0.00,0.03,0.04,0.03,0.01,0.00,0.01,normal,21

0,tcp,http,SF,199,420,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,30,32,0.00,0.00,0.00,0.00,1.00,0.

00,0.09,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,121,19,0.00,0.00,1.00,1.00,0.16,0.

06,0.00,255,19,0.07,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,166,9,1.00,1.00,0.00,0.00,0.05,0.06,

0.00,255,9,0.04,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,117,16,1.00,1.00,0.00,0.00,0.14,0.0

6,0.00,255,15,0.06,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,remote_job,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,270,23,1.00,1.00,0.00,0.00,0.09

,0.05,0.00,255,23,0.09,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,133,8,1.00,1.00,0.00,0.00,0.06,0.06,

0.00,255,13,0.05,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,205,12,0.00,0.00,1.00,1.00,0.06,0.

06,0.00,255,12,0.05,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,199,3,1.00,1.00,0.00,0.00,0.02,0.06,
0.00,255,13,0.05,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,http,SF,287,2251,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0.00,0.00,0.00,0.00,1.00,0.0
0,0.43,8,219,1.00,0.00,0.12,0.03,0.00,0.00,0.00,0.00,normal,21
0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,2,20,1.00,0.00,1.00,0.20,0.00,0.00,0.00,0.00,warezclient,15
0,tcp,name,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,233,1,1.00,1.00,0.00,0.00,0.00,0.06,0
.00,255,1,0.00,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,netbios_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,96,16,1.00,1.00,0.00,0.00,0.17,0
.05,0.00,255,2,0.01,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18
0,tcp,http,SF,300,13788,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,9,0.00,0.11,0.00,0.00,1.00,0.
00,0.22,91,255,1.00,0.00,0.01,0.02,0.00,0.00,0.00,0.00,normal,21
0,icmp,eco_i,SF,18,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,1,16,1.00,0.00,1.00,1.00,0.00,0.00,0.00,0.00,ipsweep,18
0,tcp,http,SF,233,616,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,66,255,1.00,0.00,0.02,0.03,0.00,0.00,0.02,0.00,normal,21
0,tcp,http,SF,343,1178,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,10,0.00,0.00,0.00,0.00,1.00,0.
00,0.20,157,255,1.00,0.00,0.01,0.04,0.00,0.00,0.00,0.00,normal,21
0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,223,23,1.00,1.00,0.00,0.00,0.10,0.05,0.
00,255,23,0.09,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,280,17,1.00,1.00,0.00,0.00,0.06,0.0
5,0.00,238,17,0.07,0.06,0.00,0.00,0.99,1.00,0.00,0.00,neptune,18
0,tcp,http,SF,253,11905,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,10,0.00,0.00,0.00,0.00,1.00,0
.00,0.20,87,255,1.00,0.00,0.01,0.02,0.00,0.00,0.00,0.00,normal,21

5607,udp,other,SF,147,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00
,0.00,0.00,255,1,0.00,0.85,1.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,248,2,1.00,1.00,0.00,0.00,0.01,0.06,0.0
0,255,2,0.01,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
507,tcp,telnet,SF,437,14421,0,0,0,0,0,1,3,0,0,0,0,0,1,0,0,0,1,1,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,255,25,0.10,0.05,0.00,0.00,0.53,0.00,0.02,0.16,normal,20
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,279,7,1.00,1.00,0.00,0.00,0.03,0.06,
0.00,255,13,0.05,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,http,SF,227,6588,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,22,0.00,0.00,0.00,0.00,1.00,0.
00,0.18,43,255,1.00,0.00,0.02,0.14,0.00,0.00,0.56,0.57,normal,21
0,tcp,http,SF,215,10499,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,14,14,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,241,1400,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,33,33,0.00,0.00,0.03,0.03,1.00,0
.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,12,0.00,0.00,0.00,0.00,1.00,0.00,
1.00,1,53,1.00,0.00,1.00,0.51,0.00,0.00,0.00,0.00,ipsweep,17
0,tcp,finger,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,57,16,1.00,1.00,0.00,0.00,0.28,0.07,
0.00,255,59,0.23,0.04,0.01,0.00,1.00,1.00,0.00,0.00,neptune,20
0,tcp,http,SF,303,555,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,9,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,9,255,1.00,0.00,0.11,0.01,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,1,0.00,0.00,1.00,1.00,0.50,1.00,
0.00,255,1,0.00,0.31,0.28,0.00,0.00,0.00,0.29,1.00,portsweep,20
0,udp,domain_u,SF,45,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,181,181,0.00,0.00,0.00,0.00,1
.00,0.00,0.00,255,250,0.98,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,18

1,udp,private,SF,105,147,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,41,5,0.12,0.05,0.05,0.00,0.00,0.00,0.00,0.00,normal,19
0,udp,domain_u,SF,43,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,122,202,0.00,0.00,0.00,0.00,1
.00,0.00,0.01,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,normal,18
0,tcp,supdup,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,205,17,1.00,1.00,0.00,0.00,0.08,0.0
6,0.00,255,17,0.07,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,http,SF,324,2302,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,22,28,0.00,0.00,0.00,0.00,1.00,0
.00,0.14,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,uucp_path,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,228,13,1.00,1.00,0.00,0.00,0.06,
0.07,0.00,255,13,0.05,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,20
0,tcp,ftp_data,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,50,11,1.00,1.00,0.00,0.00,0.22,0.0
8,0.00,255,63,0.25,0.02,0.01,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,Z39_50,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,262,12,1.00,1.00,0.00,0.00,0.05,0.
06,0.00,255,9,0.04,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18
2,tcp,smtp,SF,1591,372,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.00,1.00,0.0
0,1.00,181,147,0.81,0.02,0.01,0.00,0.00,0.00,0.00,0.00,normal,21
9052,udp,other,SF,146,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,2,0.00,0.00,0.00,0.00,0.50
,0.50,0.00,255,2,0.01,0.66,0.99,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,290,3006,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,11,11,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,csnet_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,108,11,1.00,1.00,0.00,0.00,0.10,0.
08,0.00,255,11,0.04,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,80,80,0.00,0.00,0.00,0.00,1.00,0.
00,0.00,255,80,0.31,0.02,0.31,0.00,0.00,0.00,0.00,0.00,teardrop,16

0,tcp,http,SF,255,861,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,44,44,0.02,0.02,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,4,38,1.00,0.00,1.00,0.18,0.00,0.00,0.00,0.00,warezclient,12

0,tcp,ftp_data,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,258,17,1.00,1.00,0.00,0.00,0.07,0.05,0.00,255,5,0.02,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,tcp,http,SF,302,498,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,10,0.00,0.00,0.00,0.00,1.00,0.00,0.00,131,255,1.00,0.00,0.01,0.04,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,61,4,0.00,0.00,1.00,1.00,0.07,0.10,0.00,255,4,0.02,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,2,0.01,0.02,0.01,0.00,0.00,0.00,0.77,0.00,teardrop,15

0,tcp,http,SF,220,1398,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,26,42,0.00,0.00,0.00,0.00,1.00,0.00,0.05,26,255,1.00,0.00,0.04,0.03,0.00,0.00,0.00,0.00,normal,21

0,udp,domain_u,SF,43,69,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,120,120,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,245,0.96,0.01,0.01,0.00,0.00,0.00,0.00,0.00,normal,18

0,udp,domain_u,SF,44,133,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,73,75,0.00,0.00,0.00,0.00,1.00,0.00,0.03,122,212,0.88,0.02,0.88,0.01,0.00,0.00,0.08,0.00,normal,21

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,15,0.00,0.00,0.00,0.00,1.00,0.00,1.00,2,46,1.00,0.00,1.00,0.26,0.00,0.00,0.00,0.00,nmap,17

0,tcp,uucp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,135,9,1.00,1.00,0.00,0.00,0.07,0.06,0.00,255,11,0.04,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,20

0,tcp,finger,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,24,12,1.00,1.00,0.00,0.00,0.50,0.08,0.00,255,59,0.23,0.04,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,udp,domain_u,SF,43,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,148,228,0.00,0.00,0.00,0.00,1
.00,0.00,0.01,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,normal,18
0,tcp,ftp_data,SF,641,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,175,48,0.25,0.02,0.25,0.04,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,206,4,0.00,0.00,1.00,1.00,0.02,0.0
6,0.00,255,4,0.02,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,175,1,0.10,0.00,0.89,1.00,0.01,1.0
0,0.00,255,1,0.00,0.84,0.00,0.00,0.07,0.00,0.62,1.00,satan,18
0,tcp,http,SF,310,2794,0,0,0,0,1,0,0,0,0,0,0,0,0,0,4,4,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,63,10,0.00,0.00,1.00,1.00,0.16,0.0
8,0.00,255,10,0.04,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,smtp,SF,696,333,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00
,0.00,109,133,0.39,0.04,0.01,0.02,0.00,0.00,0.00,0.00,normal,21
0,tcp,netbios_dgm,RSTR,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,1.00,1.00,1.
00,0.00,0.00,236,1,0.00,0.58,0.58,0.00,0.00,0.00,0.58,1.00,portsweep,14
0,tcp,http,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,52,19,1.00,1.00,0.00,0.00,0.37,0.08,0.0
0,255,67,0.26,0.02,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,netbios_dgm,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,125,7,0.00,0.00,1.00,1.00,0.
06,0.06,0.00,255,6,0.02,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,20
0,tcp,supdup,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,168,21,1.00,1.00,0.00,0.00,0.12,0.0
6,0.00,255,21,0.08,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,http,SF,221,2878,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,21,58,1.00,0.00,0.05,0.03,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,145,16,1.00,1.00,0.00,0.00,0.11,0.0
6,0.00,255,31,0.12,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18
0,icmp,urp_i,SF,181,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,2,0.00,0.00,0.00,0.00,0.67,0.67
,0.00,255,93,0.36,0.01,0.46,0.00,0.00,0.00,0.00,0.00,normal,21
0,udp,domain_u,SF,42,42,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,14,0.00,0.00,0.00,0.00,1.00,
0.00,0.21,84,83,0.99,0.02,0.01,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,138,6,1.00,1.00,0.00,0.00,0.04,0.06,
0.00,255,6,0.02,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,ftp_data,SF,12,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,44,47,0.55,0.07,0.55,0.04,0.00,0.00,0.00,0.00,normal,20
0,tcp,auth,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,276,9,1.00,1.00,0.00,0.00,0.03,0.06,0.
00,255,9,0.04,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,http,SF,210,2790,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,27,0.00,0.00,0.00,0.00,1.00,0
.00,0.11,48,255,1.00,0.00,0.02,0.09,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,321,5683,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,10,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,168,255,1.00,0.00,0.01,0.04,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,248,8,0.00,0.00,1.00,1.00,0.03,0.0
6,0.00,255,8,0.03,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,http,SF,329,3982,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,31,0.00,0.00,0.00,0.00,1.00,0.
00,0.10,24,255,1.00,0.00,0.04,0.05,0.00,0.00,0.00,0.00,normal,21
0,tcp,uucp_path,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,135,7,0.00,0.00,1.00,1.00,0.05,
0.07,0.00,255,7,0.03,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,19
0,tcp,domain,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,220,19,1.00,1.00,0.00,0.00,0.09,0.0
5,0.00,255,2,0.01,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,44,0.00,0.00,0.00,0.00,1.00,0.00,
1.00,1,95,1.00,0.00,1.00,0.51,0.00,0.00,0.00,0.00,ipsweep,15

0,tcp,http,SF,320,379,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,9,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,123,255,1.00,0.00,0.01,0.02,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,212,7,1.00,1.00,0.00,0.00,0.03,0.05,
0.00,72,7,0.10,0.11,0.01,0.00,0.97,1.00,0.00,0.00,neptune,18

0,tcp,http,SF,225,3762,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,10,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,ftp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,250,13,1.00,1.00,0.00,0.00,0.05,0.06,0.
00,255,1,0.00,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,132,15,0.00,0.00,1.00,1.00,0.11,0.
05,0.00,255,15,0.06,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

315,udp,other,SF,146,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,2,0.00,0.00,0.00,0.00,0.67,
0.67,0.00,117,2,0.02,0.54,0.78,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,247,1416,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,11,11,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,156,255,1.00,0.00,0.01,0.09,0.00,0.00,0.46,0.61,normal,21

0,tcp,http,SF,353,1483,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,9,0.00,0.00,0.00,0.00,1.00,0.0
0,0.33,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,udp,domain_u,SF,44,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,109,189,0.00,0.00,0.00,0.00,0
.99,0.02,0.01,255,254,1.00,0.01,0.01,0.00,0.00,0.00,0.00,0.00,normal,18

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,146,18,0.00,0.00,1.00,1.00,0.12,0.
05,0.00,255,10,0.04,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,csnet_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,300,6,1.00,1.00,0.00,0.00,0.02,0.0
6,0.00,255,3,0.01,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

2,tcp,smtp,SF,3065,331,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.00,1.00,0.0
0,1.00,243,119,0.48,0.02,0.00,0.02,0.01,0.02,0.00,0.00,normal,21
0,udp,other,SF,102,102,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,7,1,0.14,0.29,0.14,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,259,750,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,4,5,0.00,0.00,0.00,0.00,1.00,0.00,
0.40,4,255,1.00,0.00,0.25,0.03,0.00,0.00,0.00,0.00,normal,21
1082,udp,other,SF,147,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00
,0.00,0.00,255,2,0.01,0.42,0.86,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,129,16,1.00,1.00,0.00,0.00,0.12,0.0
5,0.00,255,12,0.05,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,http,SF,0,354,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,32,32,0.03,0.03,0.00,0.00,1.00,0.00,
0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,20
0,tcp,http,SF,309,306,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,66,255,1.00,0.00,0.02,0.02,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,1.00,1.00,1.00,0.00,1.0
0,1,247,1.00,0.00,1.00,0.15,0.00,0.00,1.00,0.68,normal,21
0,tcp,http,SF,278,7227,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,153,255,1.00,0.00,0.01,0.05,0.00,0.00,0.00,0.00,normal,21
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,82,82,0.00,0.00,0.00,0.00,1.00,0.
00,0.00,255,182,0.71,0.29,0.71,0.00,0.09,0.00,0.20,0.00,teardrop,16
0,tcp,http,SF,313,761,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,43,255,1.00,0.00,0.02,0.03,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,219,1212,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,15,15,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,242,255,1.00,0.00,0.00,0.01,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,245,9165,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,3,0.00,0.00,0.00,0.00,1.00,0.0

0,0.67,10,255,1.00,0.00,0.10,0.05,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,328,2243,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,21,21,0.00,0.00,0.00,0.00,1.00,0

.00,0.00,21,255,1.00,0.00,0.05,0.04,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,1.00,1.00,1.00,0.00,0.0

0,23,249,1.00,0.00,0.04,0.10,0.00,0.00,1.00,0.96,normal,21

0,tcp,http,SF,251,301,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0.00,0.00,0.00,0.00,1.00,0.00,

0.29,255,252,0.99,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,smtp,SF,2089,335,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,0.50,1.0

0,1.00,255,152,0.60,0.03,0.00,0.00,0.01,0.02,0.00,0.01,normal,20

0,tcp,http,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,1.00,1.00,1.00,0.00,0.0

0,6,255,1.00,0.00,0.17,0.21,0.00,0.00,1.00,0.88,normal,21

0,tcp,http,SF,237,511,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,

0.00,3,255,1.00,0.00,0.33,0.01,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,324,431,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,4,5,0.00,0.00,0.00,0.00,1.00,0.00,

0.40,107,255,1.00,0.00,0.01,0.02,0.00,0.00,0.00,0.00,normal,21

25950,tcp,private,RSTR,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,1.00,1.00,1.0

0,0.00,0.00,255,2,0.01,0.69,1.00,0.00,0.00,0.00,1.00,1.00,portsweep,15

0,tcp,Z39_50,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,110,17,0.00,0.00,1.00,1.00,0.15,0

.05,0.00,255,17,0.07,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,18

0,udp,private,SF,105,145,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,1,0.00,0.00,0.00,0.00,0.25,0

.50,0.00,255,239,0.94,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,18

0,tcp,http,SF,218,974,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,6,6,0.00,0.00,0.00,0.00,1.00,0.00,

0.00,6,255,1.00,0.00,0.17,0.02,0.00,0.01,0.00,0.00,normal,21

0,tcp,http,SF,325,2649,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,7,7,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,107,255,1.00,0.00,0.01,0.01,0.00,0.00,0.00,0.00,normal,21

0,tcp,uucp_path,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,246,20,0.00,0.00,1.00,1.00,0.0

8,0.07,0.00,255,20,0.08,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,19

0,tcp,bgp,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,272,19,0.00,0.00,1.00,1.00,0.07,0.07,

0.00,255,19,0.07,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,19

0,tcp,csnet_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,259,8,1.00,1.00,0.00,0.00,0.03,0.0

7,0.00,255,8,0.03,0.08,0.00,0.00,1.00,1.00,0.00,0.00,neptune,20

240,tcp,http,SF,328,275,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,10,0.00,0.00,0.00,0.10,1.00,0

.00,0.20,255,250,0.98,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,bgp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,294,6,1.00,1.00,0.00,0.00,0.02,0.05,0.0

0,255,8,0.03,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,18

0,tcp,finger,SF,9,138,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,

0.00,195,10,0.03,0.03,0.01,0.20,0.01,0.00,0.01,0.00,normal,21

0,udp,domain_u,SF,46,46,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,96,178,0.00,0.00,0.00,0.00,1.

00,0.00,0.02,255,254,1.00,0.01,0.01,0.00,0.00,0.00,0.00,0.00,normal,18

1,tcp,smtp,SF,1079,334,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,138,167,0.57,0.03,0.01,0.01,0.00,0.00,0.00,0.00,normal,21

0,udp,domain_u,SF,45,139,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,79,159,0.00,0.00,0.00,0.00,1

.00,0.00,0.01,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,19

26,tcp,ftp,SF,273,903,0,0,0,5,0,1,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,

0.00,176,49,0.28,0.02,0.01,0.00,0.01,0.02,0.00,0.00,normal,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,190,20,1.00,1.00,0.00,0.00,0.11,0.0

6,0.00,255,20,0.08,0.08,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

Figure 1: NSL-KDD training dataset

Source: Github (2020)

APPENDIX II

NSL-KDD TEST DATASET

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,229,10,0.00,0.00,1.00,1.00,0.04,0.06,0.00,255,10,0.04,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,136,1,0.00,0.00,1.00,1.00,0.01,0.06,0.00,255,1,0.00,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

2,tcp,ftp_data,SF,12983,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,134,86,0.61,0.04,0.61,0.02,0.00,0.00,0.00,0.00,normal,21

0,icmp,eco_i,SF,20,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,65,0.00,0.00,0.00,0.00,1.00,0.00,1.00,3,57,1.00,0.00,1.00,0.28,0.00,0.00,0.00,0.00,saint,15

1,tcp,telnet,RSTO,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,8,0.00,0.12,1.00,0.50,1.00,0.00,0.75,29,86,0.31,0.17,0.03,0.02,0.00,0.00,0.83,0.71,mscan,11

0,tcp,http,SF,267,14515,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,4,4,0.00,0.00,0.00,0.00,1.00,0.00,0.00,155,255,1.00,0.00,0.01,0.03,0.01,0.00,0.00,0.00,normal,21

0,tcp,smtp,SF,1022,387,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,3,0.00,0.00,0.00,0.00,1.00,0.00,1.00,255,28,0.11,0.72,0.00,0.00,0.00,0.00,0.72,0.04,normal,21

0,tcp,telnet,SF,129,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.01,0.01,0.02,0.02,guess_passwd,15

0,tcp,http,SF,327,467,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,33,47,0.00,0.00,0.00,0.00,1.00,0.00,0.04,151,255,1.00,0.00,0.01,0.03,0.00,0.00,0.00,0.00,normal,21

0,tcp,ftp,SF,26,157,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,52,26,0.50,0.08,0.02,0.00,0.00,0.00,0.00,0.00,guess_passwd,7

0,tcp,telnet,SF,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,128,0.50,0.01,0.00,0.00,0.00,0.00,0.66,0.32,mscan,9

0,tcp,smtp,SF,616,330,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.00,1.00,0.00
,1.00,255,129,0.51,0.03,0.00,0.00,0.00,0.00,0.33,0.00,normal,18
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,111,2,0.00,0.00,1.00,1.00,0.02,0.0
7,0.00,255,2,0.01,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,telnet,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,120,120,1.00,1.00,0.00,0.00,1.00,0.0
0,0.00,235,171,0.73,0.07,0.00,0.00,0.69,0.95,0.02,0.00,neptune,18
37,tcp,telnet,SF,773,364200,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,38,73,0.16,0.05,0.03,0.04,0.00,0.77,0.00,0.07,normal,14
0,tcp,http,SF,350,3610,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,71,255,1.00,0.00,0.01,0.04,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,213,659,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,24,24,0.00,0.00,0.00,0.00,1.00,0.
00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,246,2090,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,16,16,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,35,255,1.00,0.00,0.03,0.05,0.00,0.00,0.00,0.00,normal,21
0,udp,private,SF,45,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,505,505,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,normal,15
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,204,18,0.00,0.00,1.00,1.00,0.09,0.
07,0.00,255,18,0.07,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,ldap,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,118,19,0.00,0.00,1.00,1.00,0.16,0.05,
0.00,255,19,0.07,0.05,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,pop_3,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1.00,1.00,0.00,0.00,1.00,0.00,0.0
0,255,87,0.34,0.01,0.01,0.00,1.00,1.00,0.00,0.00,mscan,18
0,tcp,http,SF,196,1823,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,17,17,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,277,1816,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,17,18,0.00,0.00,0.00,0.00,1.00,0
.00,0.11,36,255,1.00,0.00,0.03,0.02,0.00,0.00,0.00,0.00,normal,21

0,tcp,courier,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,116,8,0.00,0.00,1.00,1.00,0.07,0.0
7,0.00,255,8,0.03,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,18

0,tcp,discard,RSTO,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,273,13,0.00,0.00,1.00,1.00,0.05
,0.06,0.00,255,13,0.05,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,20

0,tcp,http,SF,294,6442,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,22,46,0.00,0.00,0.00,0.00,1.00,0
.00,0.11,180,255,1.00,0.00,0.01,0.01,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,300,440,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,7,7,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,icmp,ecr_i,SF,520,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,46,59,1.00,0.00,1.00,0.14,0.00,0.00,0.00,0.00,smurf,18

0,udp,private,SF,54,51,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,255,255,1.00,0.00,0.83,0.00,0.00,0.00,0.00,0.00,normal,17

805,tcp,http,RSTR,76944,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,12,12,0.00,0.00,1.00,1.00,1.
00,0.00,0.00,241,238,0.99,0.01,0.00,0.00,0.00,0.00,0.07,0.07,apache2,14

0,tcp,smtp,SF,720,281,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.00,1.00,0.00
,1.00,158,82,0.52,0.06,0.01,0.00,0.01,0.00,0.00,0.00,normal,21

0,tcp,http,SF,301,19794,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,15,15,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,20,255,1.00,0.00,0.05,0.02,0.05,0.00,0.00,0.00,normal,21

0,udp,private,SF,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,40,3,0.00,0.00,0.00,0.00,0.08,0.38,
0.00,255,3,0.01,0.58,0.99,0.00,0.00,0.00,0.01,0.00,normal,1

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,483,1,0.05,0.00,0.92,1.00,0.00,1.0
0,0.00,255,1,0.00,1.00,0.00,0.00,0.00,0.00,0.96,1.00,satan,19

0,tcp,imap4,RSTO,0,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,1.00,1.00,0.50,1.0
0,1.00,185,59,0.24,0.03,0.01,0.03,0.01,0.00,0.89,0.95,mscan,14
0,tcp,http,SF,209,12894,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,11,11,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
8,tcp,ftp,SF,220,688,0,0,0,4,0,1,0,0,0,0,4,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0
.00,53,27,0.51,0.08,0.02,0.00,0.00,0.00,0.00,0.00,buffer_overflow,1
0,udp,domain_u,SF,43,71,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,113,113,0.00,0.00,0.00,0.00,1
.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,20
0,tcp,ftp_data,SF,88382,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,15,15,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,203,114,0.38,0.01,0.38,0.02,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,144,8,1.00,1.00,0.00,0.00,0.06,0.06,
0.00,255,8,0.03,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,http,SF,277,4968,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,13,13,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,13,255,1.00,0.00,0.08,0.01,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,321,2715,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,29,37,0.03,0.05,0.00,0.00,1.00,0
.00,0.08,29,255,1.00,0.00,0.03,0.04,0.03,0.00,0.00,0.00,normal,21
0,tcp,http,SF,335,3228,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,49,50,0.00,0.00,0.00,0.00,1.00,0
.00,0.04,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,mtp,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,266,8,0.00,0.00,1.00,1.00,0.03,0.06,0
.00,255,8,0.03,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,20
0,tcp,http,SF,234,3236,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,21,0.00,0.00,0.00,0.00,1.00,0.
00,0.10,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,systat,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,281,6,1.00,1.00,0.00,0.00,0.02,0.06,0
.00,255,6,0.02,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19

0,tcp,http,SF,54540,8314,0,0,0,2,0,1,1,0,0,0,0,0,0,0,0,0,4,24,0.00,0.00,0.00,0.00,1.00,0.00,0.08,255,250,0.98,0.01,0.00,0.00,0.00,0.00,0.06,0.06,back,12

0,tcp,telnet,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,12,1.00,0.33,0.00,0.67,1.00,0.00,0.92,91,86,0.34,0.03,0.01,0.03,1.00,1.00,0.00,0.00,mscan,18

0,udp,private,SF,46,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,68,68,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.26,0.00,0.00,0.00,0.00,0.00,normal,14

0,tcp,http,SF,318,488,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,16,0.00,0.00,0.00,0.00,1.00,0.00,0.12,3,156,1.00,0.00,0.33,0.04,0.00,0.01,0.00,0.00,normal,21

0,tcp,http,SF,256,4926,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,41,0.00,0.00,0.00,0.00,1.00,0.00,0.07,255,241,0.95,0.02,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,iso_tsap,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,239,7,0.00,0.00,1.00,1.00,0.03,0.06,0.00,255,7,0.03,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,20

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,201,3,0.00,0.00,1.00,1.00,0.01,0.08,0.00,255,3,0.01,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

282,tcp,ftp,SF,160,599,0,0,0,2,0,1,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,37,0.15,0.02,0.00,0.00,0.00,0.00,0.44,0.00,warezmaster,13

0,tcp,http,SF,229,509,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,21,255,1.00,0.00,0.05,0.04,0.00,0.00,0.00,0.00,normal,21

0,tcp,imap4,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,280,8,0.00,0.00,1.00,1.00,0.03,0.06,0.00,255,8,0.03,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,18

0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,9,9,1.00,0.00,0.11,0.00,0.00,0.00,0.00,0.00,snmpgetattack,3

7428,tcp,telnet,SF,0,44,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,217,0.85,0.03,0.00,0.00,0.33,0.39,0.12,0.06,processtable,18

0,tcp,http,SF,282,396,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
4,tcp,pop_3,SF,28,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,255,249,0.98,0.02,0.00,0.00,0.00,0.00,0.00,0.00,guess_passwd,18
280,tcp,ftp_data,SF,283618,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,7,10,0.86,0.29,0.86,0.20,0.00,0.00,0.00,0.00,warezmaster,9
0,tcp,http,SF,233,2239,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,255,197,0.77,0.02,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,257,8712,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,20,0.00,0.00,0.00,0.00,1.00,0.
00,0.20,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,other,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,412,1,0.15,0.00,0.83,1.00,0.00,1.00,
0.00,255,1,0.00,0.96,0.00,0.00,0.10,0.00,0.89,1.00,satan,17
0,tcp,http,SF,214,645,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,30,30,0.00,0.00,0.00,0.00,1.00,0.
00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,telnet,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,6,0.00,0.00,1.00,1.00,0.25,1.00,0.
83,106,85,0.29,0.05,0.01,0.02,0.02,0.00,0.91,0.67,mscan,15
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,109,13,0.00,0.00,1.00,1.00,0.12,0.
06,0.00,255,13,0.05,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,smtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,114,1,1.00,1.00,0.00,0.00,0.01,0.06,0.
00,255,1,0.00,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,20
0,tcp,http,SF,203,651,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,19,19,0.00,0.00,0.00,0.00,1.00,0.
00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,216,14,0.00,0.00,1.00,1.00,0.06,0.
06,0.00,255,14,0.05,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,smtp,SF,0,83,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.00,1.00,0.00,1.0
0,61,146,0.80,0.07,0.02,0.01,0.00,0.00,0.00,0.01,normal,21
0,tcp,http,SF,307,484,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,10,255,1.00,0.00,0.10,0.05,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,320,1667,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,30,0.00,0.00,0.00,0.00,1.00,0.
00,0.17,85,253,1.00,0.00,0.01,0.04,0.00,0.00,0.00,0.00,normal,21
0,tcp,csnet_ns,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,251,3,0.00,0.00,1.00,1.00,0.01,0.
06,0.00,255,3,0.01,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,19
0,tcp,http,SF,215,430,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,40,255,1.00,0.00,0.03,0.04,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,358,3691,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,11,11,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,81,255,1.00,0.00,0.01,0.03,0.00,0.00,0.00,0.00,normal,21
8092,tcp,telnet,SF,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,255,67,0.26,0.03,0.00,0.00,0.00,0.01,0.11,0.19,processtable,8
0,tcp,http,SF,270,1332,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,7,255,1.00,0.00,0.14,0.04,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,337,283,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,24,29,0.00,0.00,0.00,0.00,1.00,0.
00,0.10,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,267,19,0.01,0.00,0.99,1.00,0.07,0.
06,0.00,255,19,0.07,0.06,0.00,0.00,0.01,0.00,0.99,1.00,neptune,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,231,6,0.00,0.00,1.00,1.00,0.03,0.0
7,0.00,255,6,0.02,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,icmp,ecr_i,SF,1480,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,1,25,1.00,0.00,1.00,0.52,0.00,0.00,0.00,0.00,pod,17

0,tcp,http,SF,219,1310,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,4,4,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,4,251,1.00,0.00,0.25,0.08,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,258,1012,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,26,26,0.00,0.00,0.00,0.00,1.00,0

.00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,324,1093,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,8,0.00,0.00,0.00,0.00,1.00,0.0

0,0.38,69,255,1.00,0.00,0.01,0.01,0.00,0.00,0.00,0.00,normal,21

0,tcp,finger,RSTO,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,58,19,0.00,0.00,1.00,1.00,0.33,0.

07,0.00,255,62,0.24,0.11,0.00,0.00,0.00,0.00,1.00,1.00,neptune,19

0,tcp,pop_3,RSTO,0,36,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0.00,0.00,1.00,1.00,0.33,1.0

0,1.00,189,60,0.24,0.03,0.01,0.03,0.00,0.00,0.88,0.98,mscan,15

0,tcp,http,SF,236,10503,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.

00,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,ftp_data,SF,1699,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.

00,0.00,36,7,0.19,0.11,0.19,0.00,0.00,0.00,0.00,0.00,normal,20

0,tcp,http,SF,327,910,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,14,15,0.00,0.00,0.00,0.00,1.00,0.

00,0.13,14,255,1.00,0.00,0.07,0.03,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,303,2789,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,103,4,0.00,0.00,1.00,1.00,0.04,0.0

6,0.00,255,4,0.02,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,other,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,1.00,1.00,1.00,0.00,0.

00,255,13,0.05,0.02,0.00,0.00,0.00,0.00,0.06,1.00,httptunnel,17

3,tcp,pop_3,SF,28,93,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,

0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,guess_passwd,17

0,tcp,smtp,SF,0,83,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.0
0,110,25,0.23,0.13,0.01,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,http,SF,282,4532,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,6,6,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,96,255,1.00,0.00,0.01,0.02,0.00,0.00,0.00,0.00,normal,21
0,udp,domain_u,SF,44,113,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,5,0.00,0.00,0.00,0.00,1.00,
0.00,0.40,255,253,0.99,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,SH,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1.00,1.00,0.00,0.00,1.00,0.00,0.
00,30,1,0.03,1.00,1.00,0.00,1.00,1.00,0.00,0.00,nmap,18
0,tcp,uucp,RSTO,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,119,7,0.00,0.00,1.00,1.00,0.06,0.0
7,0.00,255,7,0.03,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,20
0,tcp,other,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,497,1,0.00,1.00,0.99,0.00,0.00,1.00,0.
00,255,1,0.00,1.00,0.00,0.00,0.01,1.00,0.99,0.00,satan,20
0,tcp,ftp,SF,26,157,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.
00,155,71,0.46,0.03,0.01,0.00,0.00,0.00,0.00,0.00,guess_passwd,7
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,217,6,0.00,0.00,1.00,1.00,0.03,0.0
7,0.00,255,6,0.02,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,106,15,0.00,0.00,1.00,1.00,0.14,0.
08,0.00,255,15,0.06,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
1,tcp,smtp,SF,995,330,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00
,0.00,62,42,0.66,0.05,0.02,0.05,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,239,12,0.00,0.00,1.00,1.00,0.05,0.
07,0.00,255,12,0.05,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
31,tcp,telnet,SF,197,1608,0,0,0,1,0,1,1,0,0,1,2,1,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0
.00,0.00,248,32,0.13,0.03,0.00,0.00,0.00,0.00,0.00,0.00,ps,0

0,tcp,http,SF,197,1830,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,26,229,1.00,0.00,0.04,0.05,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,117,10,0.00,0.00,1.00,1.00,0.09,0.

06,0.00,255,10,0.04,0.05,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

0,tcp,http,SF,230,843,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,25,25,0.00,0.00,0.00,0.00,1.00,0.

00,0.00,25,255,1.00,0.00,0.04,0.06,0.00,0.00,0.00,0.00,normal,21

0,tcp,http,SF,161,14086,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.

00,0.00,151,151,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,228,15,0.00,0.00,1.00,1.00,0.07,0.

07,0.00,255,15,0.06,0.08,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21

8080,tcp,telnet,SF,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,255,69,0.27,0.03,0.00,0.00,0.00,0.01,0.11,0.19,processtable,8

8124,tcp,telnet,SF,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.0

0,0.00,255,60,0.24,0.04,0.00,0.00,0.00,0.02,0.11,0.22,processtable,8

0,tcp,http,SF,224,278,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,5,0.00,0.00,0.00,0.00,1.00,0.00,

0.40,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,259,20,1.00,1.00,0.00,0.00,0.08,0.0

7,0.00,255,20,0.08,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

0,tcp,http,SF,321,372,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,16,17,0.00,0.00,0.00,0.00,1.00,0.

00,0.12,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21

0,icmp,ecr_i,SF,1032,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,173,173,0.00,0.00,0.00,0.00,1.0

0,0.00,0.00,255,254,1.00,0.01,1.00,0.00,0.00,0.00,0.00,0.00,smurf,21

0,tcp,whois,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,282,17,0.00,0.00,1.00,1.00,0.06,0.0

6,0.00,255,17,0.07,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,20

0,udp,private,SF,48,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,5,5,0.00,0.00,0.00,0.00,1.00,0.00,
0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,snmpguess,15
0,tcp,http,SF,303,289,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,6,11,0.00,0.00,0.00,0.00,1.00,0.0
0,0.36,30,255,1.00,0.00,0.03,0.01,0.00,0.00,0.00,0.00,normal,21
0,icmp,ecr_i,SF,508,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0.00,0.00,0.00,0.00,1.00,
0.00,0.00,255,192,0.75,0.02,0.75,0.00,0.00,0.00,0.23,0.00,smurf,18
0,tcp,private,SH,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1.00,1.00,0.00,0.00,1.00,0.00,0.
00,80,1,0.01,1.00,1.00,0.00,1.00,1.00,0.00,0.00,nmap,19
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,132,12,0.00,0.00,1.00,1.00,0.09,0.
05,0.00,255,12,0.05,0.06,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,finger,SF,8,447,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.00,1.00,0.00,
1.00,74,10,0.04,0.31,0.01,0.20,0.01,0.00,0.00,0.00,normal,21
0,udp,domain_u,SF,43,78,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,208,207,0.00,0.00,0.00,0.00,1
.00,0.01,0.00,255,254,1.00,0.01,0.01,0.00,0.00,0.00,0.00,0.00,normal,18
0,tcp,telnet,SF,124,174,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.0
0,0.00,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.02,0.02,guess_passwd,16
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,226,2,0.03,0.00,0.97,1.00,0.01,0.0
7,0.00,255,2,0.01,0.08,0.00,0.00,0.02,0.00,0.98,1.00,neptune,21
0,tcp,http,SF,257,633,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,4,0.00,0.00,0.00,0.00,1.00,0.00,
0.50,171,255,1.00,0.00,0.01,0.03,0.00,0.00,0.00,0.00,normal,21
0,udp,domain_u,SF,42,42,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,7,12,0.00,0.00,0.00,0.00,1.00,
0.00,0.25,220,220,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,udp,domain_u,SF,42,42,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,19,0.00,0.00,0.00,0.00,1.0
0,0.00,0.11,255,255,1.00,0.00,0.01,0.00,0.00,0.00,0.00,0.00,normal,21

Figure 2: NSL-KDD test dataset
Source: Github (2020)

APPENDIX III

FINAL RESULTS

SN	PC1	PC2	PC3	PC4	PC5	PC6	PC7
5188	-1.43341	0.596405	-0.44467	-1.14996	2.509284	1.120042	-1.56106
110012	1.208092	0.588724	0.065598	-2.68224	3.190255	1.539098	-2.00343
289664	4.76409	-4.17956	-0.4102	0.516971	0.221604	0.315511	-0.40474
11733	-3.80554	0.220176	-6.16195	0.148521	-1.61079	0.749816	-0.13229
112761	4.891938	-4.20786	-0.46565	0.473168	0.239561	0.282929	-0.35332
91777	-1.45234	-0.159	1.378551	-0.01186	0.491553	-0.4992	0.621893
241358	-2.60999	0.093091	0.456637	1.193181	0.238186	-1.14077	0.08406
18724	-1.99626	-0.54995	-6.02778	-4.54614	0.817659	0.947084	1.188159
37521	-0.59901	0.710094	0.353775	-0.04459	-0.22459	0.124985	0.222975

PC8	PC9	PC10	PC11	PC12	PC13	PC14	PC15
0.849499	0.064919	-0.36444	0.981542	-0.09704	0.566431	-1.09105	-0.30802
1.376987	0.27099	-1.39076	0.896866	-0.73486	0.056956	0.346718	0.048024
0.2378	0.55475	-0.44988	-0.40682	-0.66058	0.008154	0.413063	0.305829
-0.60043	0.506227	-1.29716	0.044736	0.301274	-0.06884	-0.01851	-0.0492
0.18615	0.370871	-0.41412	-0.39389	-0.65759	0.039806	0.403971	0.2998
-0.73715	0.197757	-1.81841	-1.07062	-2.35901	1.654002	-0.73425	-0.37695
1.828881	-0.35664	1.213269	0.734716	-0.20698	0.142829	-0.01658	0.121971
-4.17533	-3.16717	8.379447	-5.16169	-2.3807	0.301356	0.105053	0.057338
-0.80542	0.485391	-2.60376	-0.95341	-2.17076	1.730463	-0.94782	-0.43489

PC16	PC17	PC18	PC19	PC20	PC21	PC22	PC23
-1.8841	-0.10106	-0.21132	0.521873	-1.12022	0.028723	-0.14176	-0.23805
0.775836	0.451745	-0.62111	-0.14419	-0.1341	0.058582	0.01413	0.108368
0.972755	0.277142	-0.23485	-0.17009	-0.07654	0.10105	-0.00301	-0.04474
-0.14783	-0.21819	-0.02033	0.109183	0.029106	0.036225	0.056989	0.027328
0.940998	0.250505	-0.23763	-0.15985	-0.08607	0.11368	0.012623	-0.04428
-1.69693	-0.06101	-0.68544	-0.83662	4.697652	-1.31443	-0.97666	-0.29589
0.250291	-0.09655	0.033966	-0.03281	0.147765	-0.03313	0.142869	0.183885
0.442796	0.67874	0.099866	-0.19613	0.368079	-0.18241	-0.07104	-0.03548
-1.98836	-0.10621	-0.77626	-0.86778	5.51174	-1.51989	-1.06084	-0.31204

PC24	PC25	PC26	PC27	PC28	PC29	PC30	PC31
-0.04097	-0.24601	0.274055	-0.02524	-0.31038	-0.5044	0.464046	0.248291
-0.06636	0.006556	0.029092	0.01416	-0.03842	-0.03363	0.093504	-0.02029
-0.08857	-0.10364	0.024803	-0.00698	-0.03687	-0.07626	0.071422	0.110993
-0.08436	-0.03596	0.087156	0.024951	-0.01998	-0.07275	0.032027	0.064439
-0.12272	-0.13227	0.02922	-0.01486	-0.03268	-0.0861	0.095472	0.118817
0.162373	0.637566	0.206886	-0.00399	0.119879	0.321275	-0.32297	0.428508
-0.06653	0.171206	0.108496	0.065695	-0.08492	-0.01409	0.028414	-0.06793
-0.16956	-0.07961	0.190489	-0.01736	-0.32991	-0.43887	0.237063	0.279467
0.106846	0.716157	0.290157	-0.0125	0.08961	0.370457	-0.38355	0.43724

PC32	PC33	PC34	PC35	PC36	PC37	PC38	PC39
0.274289	0.168795	0.196434	0.180756	0.124714	-0.03128	0.276966	-0.21958
0.013292	-0.03607	-0.02418	-0.01627	-0.01714	0.014987	0.003005	0.017857
0.108795	-0.02033	0.052084	0.02941	-0.0094	0.009499	0.022714	-0.03871
0.056664	-0.03152	0.056972	0.015423	0.008694	0.02103	0.046384	-0.00665
0.132343	-0.02636	0.055578	0.021791	-0.00643	0.008927	0.028292	-0.04374
-0.64849	-0.09869	-0.27961	-0.02805	0.31	0.020309	-0.23659	0.029365
-0.19687	-0.09783	-0.16139	-0.04988	-0.05709	0.059904	-0.14054	0.056844
0.299154	-0.02813	0.127981	0.308976	0.054738	-0.03959	0.247038	-0.0705
-0.65779	-0.14986	-0.31848	-0.0607	0.353178	0.040466	-0.20971	0.032624

PC40	PC41	PC42	PC43	PC44	PC45	PC46	PC47
0.164281	0.271315	0.298299	-0.21565	0.161457	-0.11932	0.092712	0.019399
0.001184	0.01087	0.000783	0.083221	-0.00975	-0.01198	0.015618	0.004404
-0.01296	-0.0519	0.014324	0.176439	0.007044	-0.01006	-0.0017	-0.0012
0.026593	0.072205	0.069764	0.052549	-0.01969	0.025628	0.005628	-0.00473
-0.00978	-0.03571	0.001406	0.18288	-0.00516	-0.01508	0.000759	-0.00132
-0.26532	0.170807	-0.025	0.047128	0.05009	-0.11704	-0.08584	0.006675
-0.05366	0.010414	-0.0402	0.038305	-0.02805	-0.02097	0.036438	0.009034
0.01809	0.126591	0.353941	-0.34637	0.132045	0.196084	-0.2348	-0.13636
-0.28506	0.232716	0.003749	0.022029	0.034817	-0.11401	-0.09436	0.010222

PC48	PC49	PC50	PC51	PC52	PC53	PC54	PC55
0.01545	0.013883	-0.01929	-0.09379	0.033215	0.048525	-0.0909	-0.05882
0.00741	0.002983	-0.0018	0.006215	0.014866	-0.01186	-0.01966	-0.01573
0.006924	-0.00822	-0.00056	-0.01273	-0.00421	0.021861	-0.00565	0.002936
0.003232	0.00501	-0.00183	-0.00391	0.025313	-0.01366	-0.01784	0.004542
0.007851	-0.00818	-0.00074	-0.0133	0.005196	0.025573	-0.00801	0.004914
0.007316	-0.05436	-0.0034	-0.00966	-0.0073	-0.24796	0.175255	-0.0329
-0.00609	0.001456	0.00142	0.034947	-0.00052	-0.02172	-0.0286	-0.00334
-0.00628	0.013532	0.032552	-0.03622	-0.14356	-0.03896	-0.00622	-0.03715
0.001772	-0.04338	-0.00754	-0.00837	0.043197	-0.31472	0.191177	-0.02769

PC56	PC57	PC58	PC59	PC60	PC61	PC62	PC63
-0.10556	-0.20741	-0.00829	-0.19157	-0.03852	-0.02104	-0.05698	-0.06674
-0.01319	0.002367	-0.01418	0.000238	-0.0067	-0.00202	-0.0009	0.010973
0.046428	0.049624	-0.01655	0.013147	-0.00464	-0.01317	-0.03436	0.027717
-0.00928	-0.00951	-0.01931	-0.02858	-0.01658	0.002502	0.014562	0.004828
0.039688	0.045607	-0.02449	0.011386	-0.00897	-0.00885	-0.02353	0.03172
-0.18207	0.008792	-0.00169	0.052061	0.03718	-0.05403	0.061843	0.022992
-0.02392	0.020733	0.003774	0.004646	0.008505	0.010498	0.0128	0.020946
0.006501	-0.00617	0.034188	-0.00812	-0.00269	-0.03655	-0.13021	-0.02872
-0.20914	-0.00162	0.003402	0.026731	0.060524	-0.05699	0.105273	0.012538

PC64	PC65	PC66	PC67	PC68	PC69	PC70	PC71
0.026465	-0.03984	0.131631	-0.431	-0.09089	0.145141	-0.04534	-0.02046
0.003148	0.039914	0.005963	0.04663	0.017481	-0.04266	0.025823	-0.00114
0.037506	0.086355	-0.02088	0.115364	0.046035	-0.08443	0.030532	0.0008
0.017342	0.070286	0.009329	-0.01818	0.005931	0.017778	0.007867	-0.01791
0.038085	0.093789	-0.0201	0.118175	0.050006	-0.08391	0.034083	0.001436
-0.02295	0.134086	-0.19176	0.102926	0.074559	-0.05577	-0.00405	0.007944
-0.00428	0.022551	-0.01414	0.021962	0.00581	0.013413	0.00299	-0.01391
0.019974	-0.003	-0.0029	-0.01204	-0.0191	-0.06942	-0.00806	-0.01469
-0.0362	0.114101	-0.18693	0.076515	0.073868	-0.01479	-0.01284	0.00777

PC72	PC73	PC74	PC75	PC76	PC77	PC78	PC79
-0.05586	-0.00547	0.232443	-0.01008	0.022562	-0.32387	-0.23442	1.0158
0.009557	-0.00674	-0.05749	0.000146	0.004887	0.102742	0.022052	-0.25433
0.000616	-0.00571	-0.11614	-0.00223	0.01973	0.178398	0.029836	-0.51407
-0.00064	0.016878	0.00158	0.000748	-0.01906	-0.02431	0.004828	-0.01447
0.002349	-0.00636	-0.11437	0.000229	0.015693	0.172527	0.034322	-0.52383
0.012066	0.071227	-0.00677	-0.00635	-0.05522	-0.09755	0.019701	-0.12645
0.005417	0.013893	-0.00031	-0.00818	-0.02002	-0.03588	0.014368	0.013182
-0.00198	-0.01928	-0.05049	-0.00889	0.037035	0.107675	-0.02587	-0.10588
0.007949	0.099113	0.030275	-0.01287	-0.07317	-0.18672	0.016234	0.001142

PC80	PC81	PC82	PC83	PC84	PC85
0.187921	0.042053	-0.24301	-0.06745	0.331163	-0.51075
-0.10798	-0.05209	-0.6451	0.110284	0.252581	-0.59601
-0.15209	-0.05794	-0.81644	0.267012	0.230452	-1.01476
-0.0347	-0.0068	-0.04942	-0.01854	0.010824	-0.13111
-0.15292	-0.06084	-0.81933	0.265786	0.247251	-1.06038
0.03126	-0.01655	-0.05213	-0.04446	-0.04408	0.168358
0.000854	-0.00846	-0.05918	0.0466	0.036568	-0.05721
-0.05275	-0.03255	-0.58336	0.127527	0.312746	-0.14551
0.066488	0.004521	0.308484	-0.03548	-0.19775	0.293882

PC86	PC87	PC88	class	Predicted
0.21299	-0.04361	1.33562	normal	normal
PC86	PC87	PC88	class	Predicted
0.21299	-0.04361	1.33562	normal	normal
PC86	PC87	PC88	class	Predicted
0.21299	-0.04361	1.33562	normal	normal
PC86	PC87	PC88	class	Predicted
0.21299	-0.04361	1.33562	normal	normal
PC86	PC87	PC88	class	Predicted
0.21299	-0.04361	1.33562	normal	normal

			1.34416		0.44784	1.59974	6.40775
93402	-2.00847	-0.32392	7	-0.87271	8	9	2
		0.05118			3.63668	1.01874	
313978	-0.38632	9	-0.43239	-3.18599	1	6	-1.24182
			2.34759	1.74233	1.10589	1.54867	
103367	-1.24601	-0.03116	3	9	1	8	-1.07418
	4.91758			0.51337	0.21585	0.32708	
165945	3	-4.20257	-0.43597	3	8	4	-0.41618
		0.33763				1.12225	
1011	-3.76339	4	-6.36657	0.88681	-1.86574	5	-0.72474
		0.22910		0.94729	0.16912		0.05378
124722	-2.23267	3	0.52391	5	3	-1.04238	1
			0.39503	1.05952	0.26900		0.22119
323856	-2.49873	0.16529	6	3	6	-1.2546	4
	5.28379	3.64881					1.47950
80180	9	8	-0.69371	-0.54698	-1.47147	-1.30429	8
	4.96138			0.54299	0.20326		
213785	1	-4.19194	-0.42442	1	6	0.31619	-0.42019
		0.44740		0.24380		1.16001	
58917	-2.39428	1	-4.31487	6	-1.26502	7	-0.92368

0.15138				0.11031			
3	0.35398	-1.47465	-1.28346	1	-0.13034	-1.33129	-3.19277
0.58811	0.63857		0.33377			0.52049	0.25820
9	4	-0.3355	9	-0.69904	-0.02105	5	4
				0.49002	0.24465		0.09969
-1.5806	-0.78028	-0.34204	-1.09888	3	7	-0.01601	2
0.24255	0.40820				0.04399	0.39714	0.29800
6	3	-0.43496	-0.37666	-0.6552	1	6	4
	0.36916			0.40814			0.00139
-0.09987	3	-1.36886	0.56621	8	-0.03373	-0.01574	2
1.84116		0.95665	0.57448		0.05455		0.02316
9	-0.46042	7	9	-0.28286	1	-0.07059	5
1.78802		1.02889	0.76549		0.15863	0.01797	
7	-0.66519	7	7	-0.27148	4	6	0.06354
			1.25965	3.03521		0.34088	0.17936
-0.43894	-1.67514	0.87093	2	7	-0.34689	8	2
0.27578					0.05509	0.39358	0.28839
9	0.36633	-0.50455	-0.33665	-0.6536	8	3	2
0.21454							
1	-0.02462	-0.43128	-0.00513	-0.00431	-0.04612	-0.20231	-0.19726

0.916946	0.153092	-0.23475	-0.98576	-0.42282	0.263631	0.060319	-1.11357
1.033567	0.145004	-0.17851	-0.11793	-0.09077	0.080737	0.097974	0.082656
0.035288	0.045358	0.085788	-0.18157	-0.14968	0.037235	-0.42814	-0.60339
0.943952	0.26282	-0.25309	-0.16593	-0.07795	0.106687	0.005631	-0.04429
-0.03282	-0.21975	0.075456	0.018066	0.064638	0.028902	0.061766	-0.00929
0.119043	0.012262	-0.05259	0.000687	0.184746	-0.03645	0.141669	0.200046
0.176903	-0.04569	-0.00608	-0.02805	0.049776	0.002043	0.160566	0.211681
0.233459	0.093463	-2.2807	0.008045	1.28981	-0.99265	0.832372	-0.51714
0.927868	0.268222	-0.27246	-0.16203	-0.09826	0.113433	0.007685	-0.0408
-0.27521	0.146527	-0.47978	0.114341	-0.49198	0.111577	-0.0347	0.029222

-1.25695	-3.06764	-1.05079	-0.32325	-5.40979	0.098571	0.271482	-3.4883
-0.16237	-0.02926	0.063385	-0.00627	-0.04007	-0.05525	0.079661	0.019251
0.272412	-0.4111	-0.55079	-0.32469	0.188487	0.344298	-0.4066	0.020479
-0.11247	-0.12254	0.027896	-0.01294	-0.03326	-0.08116	0.090601	0.114726
-0.10038	-0.08251	0.023081	-0.02282	-0.01289	-0.03774	0.015703	0.044594
-0.06361	0.146737	0.097958	0.072015	-0.07619	-0.04275	0.061436	-0.06552
-0.06683	0.162442	0.104216	0.074473	-0.09043	-0.02655	0.035923	-0.06256
-2.44368	-0.55583	-3.34314	-3.53173	0.512504	-0.45322	-0.80259	0.860898
-0.11549	-0.12668	0.028635	-0.01222	-0.03314	-0.08414	0.097722	0.115323
0.091286	-0.02785	0.038211	0.072612	-0.0034	-0.09908	0.098686	0.000984

		0.13177			0.12129		0.71218
-0.99719	-1.54025	5	-1.16587	-2.30575	5	-0.86996	3
							0.01361
-0.00905	-0.08556	-0.03612	-0.02118	-0.01622	0.01284	-0.01876	2
0.33666	0.20551	0.45789	0.05614	0.18971		0.39878	
7	4	1	6	1	-0.30063	8	-0.16563
0.12521		0.05251	0.02141			0.02579	
9	-0.02374	2	4	-0.00677	0.00961	1	-0.04316
		0.06436	0.01937	0.01148			
0.14307	-0.03166	1	4	5	-0.00382	0.07905	-0.01994
					0.05738		0.06450
-0.13375	-0.06263	-0.12352	-0.03014	-0.0697	8	-0.12623	9
					0.06134		0.06623
-0.18076	-0.11119	-0.15403	-0.03813	-0.06371	7	-0.13239	9
0.83491	0.21330	1.86817		4.07285	2.16563		0.67641
7	4	9	-3.05863	2	2	-2.45326	4
0.12782		0.05312	0.01999		0.01064		
1	-0.02457	9	2	-0.00717	9	0.0265	-0.04342
0.00645		0.07862	0.01568		0.03312		
7	0.10611	8	3	-0.01605	1	0.01752	-0.01517

					0.13726		0.05408
-0.14682	-0.21911	-0.1208	-0.12267	-0.19848	1	-0.00879	7
	0.00539		0.09101			0.01327	
-0.01809	8	-0.03082	4	-0.03039	-0.00612	5	-0.01447
0.10158		0.03245		0.12356			
4	-0.13109	8	-0.00696	2	-0.02103	-0.08404	-0.01186
		0.00380	0.17691				
-0.00977	-0.03749	9	3	-0.0023	-0.01445	8.58E-05	-0.00055
0.02474	0.04753	0.04981			0.02398	0.00368	
5	1	9	0.03394	-0.02055	7	9	-0.00475
	0.00401		0.04368			0.04865	0.01565
-0.03331	8	-0.01919	5	-0.01523	-0.02473	7	4
	0.00451		0.09056			0.03518	0.00321
-0.05573	5	-0.03169	1	-0.03491	-0.01244	8	8
		2.28679					0.10681
-0.98313	-4.11546	3	-0.79452	0.46845	-0.90705	-1.06202	8
		0.00254	0.18078			0.00105	
-0.00888	-0.03348	6	2	-0.00472	-0.0156	4	-0.00058
0.08299	0.04815	0.05980		0.02588	0.00499		0.02807
3	6	5	-0.0358	4	3	0.01582	7

	0.03765		0.38166	0.15262			
-0.03533	7	-0.00362	1	1	-0.0248	-0.20366	-0.1477
0.00679			0.00289	0.01357			
4	-0.00724	-0.00034	9	2	-0.00729	-0.02671	-0.0047
				0.03089	0.00547	0.07873	0.00242
-0.00047	-0.00015	0.00199	-0.10171	7	5	6	3
0.00761				0.00391	0.02481		0.00398
3	-0.00758	-0.00072	-0.01263	8	6	-0.00709	2
	0.00892			0.02169	0.01059		0.01702
-0.00099	4	-0.00132	-0.00443	3	4	-0.01045	9
	0.00451	0.00225	0.03590				
-0.00861	9	8	4	-0.01284	-0.01069	-0.03319	-0.02008
		0.00213	0.03297				0.00098
-0.00514	-0.00048	1	1	-0.00513	-0.02122	-0.03591	3
		0.07498			0.51488	0.06260	0.18431
0.06151	-0.00977	3	-0.24582	-0.51039	5	6	7
0.00800				0.00579	0.02534		0.00430
3	-0.00769	-0.00078	-0.01277	2	6	-0.0083	2
			0.00287	0.00590	0.01595		
-0.00075	0.0057	-0.00593	7	8	7	-0.01358	-0.03957

					0.17302	0.18301	
-0.01733	-0.06009	-0.02573	-0.15096	-0.13476	6	7	0.18029
	0.02521					0.00231	0.03429
0.0429	7	-0.03159	-0.00113	-0.01325	-0.00474	3	5
0.11705				0.01546			
9	-0.07748	-0.01535	-0.00569	1	-0.0455	-0.00998	-0.03547
0.03878	0.04599						
4	2	-0.02185	0.01106	-0.0075	-0.01009	-0.02555	0.02961
					0.00429	0.01198	
-0.00907	-0.00978	-0.01027	-0.02201	-0.01259	6	8	-0.00485
	0.02233	0.01163	0.01666	0.00611	0.00950	0.00214	0.01166
-0.06247	1	7	9	8	4	5	9
	0.01721		0.00808		0.01539	0.00974	0.02790
-0.01544	7	-0.00671	3	-0.00452	2	8	2
0.92669	4.79008	1.38131	2.17472	5.86201		2.71549	4.72027
7	4	7	3	9	-1.84729	7	5
0.03692	0.04354		0.01033				0.03045
3	2	-0.02406	1	-0.00919	-0.00868	-0.02345	8
		0.01377					
-0.0483	-0.03103	9	-0.01511	-0.0112	0.0024	-0.01357	-0.02411

-0.02283	-0.04117	0.32673	0.185074	0.078244	-0.13912	0.129997	
0.021302	0.089144	-0.00487	0.086259	0.030044	-0.03967	0.024926	
0.01522	0.024208	0.017749	-0.02315	0.033763	-0.04966	0.015726	
0.037248	0.088557	-0.01921	0.113988	0.047821	-0.08247	0.032677	
0.007953	0.00921	0.014574	-0.01343	-0.00321	0.013434	0.000731	
-0.013	0.000387	-0.01399	0.022538	0.001467	-0.00207	0.006419	
0.001388	0.042752	-0.0187	0.050749	0.012703	-0.00411	0.009233	
-3.13074	-2.81353	-2.9511	-0.48083	-0.15682	-0.50776	2.736617	
0.037634	0.091431	-0.01901	0.114039	0.048765	-0.08268	0.033812	
-0.00469	0.017142	0.027395	-0.11264	-0.02454	0.036674	0.004331	

0.013557	0.065301	-0.01261	0.038651	-0.28163	-0.01561	0.01736	0.388389
-0.01046	0.008455	0.0076	-0.07861	-0.00363	-0.00235	0.093733	0.029018
0.035364	-0.01856	-0.01438	0.011729	0.017141	0.055743	0.116237	-0.03273
0.001496	0.001561	-0.00668	-0.11259	-7.22E-05	0.016875	0.171387	0.031702
-0.00789	0.000474	0.003255	0.001342	0.002986	-0.00921	-0.01358	0.008073
-0.00379	0.00828	-0.00076	-0.00218	-0.00675	-0.00367	0.000275	0.014535
-0.01652	0.012102	0.013505	-0.02175	-0.00887	-0.02367	-0.00996	0.037569
0.304367	2.813478	-0.81805	-4.28553	0.676899	-2.15346	-3.02031	-0.29966
0.001348	0.002249	-0.00681	-0.1121	0.000444	0.015691	0.170377	0.033434
-0.01253	-0.00844	-0.00941	0.037057	0.004995	-0.00193	-0.00671	-0.0523

0.310078	-0.02983	-0.03452	-0.16805	-2.01771	0.07269
-0.35292	-0.11713	-0.03231	-0.58415	0.235833	0.311408
-0.20904	-0.00894	0.072983	0.497691	-0.0434	-0.31022
-0.51221	-0.15051	-0.0592	-0.81376	0.262462	0.239644
-0.02173	-0.01084	0.004634	0.040834	-0.01029	-0.05564
0.021011	-0.00393	-0.02859	-0.14757	-0.00588	0.056306
-0.07926	-0.0244	-0.02309	-0.14755	0.03957	0.054862
2.457507	0.497404	0.028088	1.125286	-0.06559	0.115585
-0.51425	-0.15186	-0.06111	-0.82355	0.258799	0.24499
0.216915	-0.02044	-0.03331	-0.27421	-0.14254	0.038104

-1.15699	-1.31997	-0.3013	0.648507	DOS	DOS
-0.78918	0.318157	0.247363	1.591094	U2R	U2R
0.35957	-0.15293	-0.18622	0.305336	DOS	DOS
-1.04296	0.264488	0.574554	-0.16736	PROBE	PROBE
-0.06109	0.003275	0.002845	-0.08558	DOS	DOS
0.113992	-0.0383	0.031037	-0.11301	normal	normal
-0.15854	0.002336	0.038658	-0.36615	U2R	U2R
1.159934	-0.1756	-0.32748	1.88739	normal	normal
-1.05705	0.266951	0.578012	-0.19794	R2L	R2L
0.17542	0.019401	-0.09956	0.567177	normal	normal

APPENDIX IV

SYSTEM DEVELOPMENT (PYTHON) CODE


```
{
 "cells": [
  {
   "cell_type": "code",
   "execution_count": 1,
   "id": "3ad25937-bea9-457e-ba87-4b00dd451d35",
   "metadata": {
    "tags": []
   },
   "outputs": [],
   "source": [
    "import numpy as np\n",
    "import pandas as pd\n",
    "import seaborn as sns\n",
    "from scipy.io import arff\n",
    "import matplotlib.pyplot as plt\n",
    "from sklearn.ensemble import VotingClassifier\n",
    "pd.set_option('display.max_columns', 200)"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "728bad7d-fa2e-4859-a5ca-2a47add33261",
```

```
"metadata": {
 "tags": []
},
"source": [
 "# Loading the Dataset"
]
},
{
"cell_type": "code",
"execution_count": 2,
"id": "f5090ad1-739f-4330-945a-6ecc6ebadd05",
"metadata": {
 "tags": []
},
"outputs": [],
"source": [
 "#Load the training dataset\n",
 "data = pd.read_csv(\"KDDTrain+.txt\")\n",
 "data_2, meta = arff.loadarff('KDDTrain+.arff')\n",
 "data_2 = pd.DataFrame(data_2)\n",
 "\n",
 "\n",
 "# Load the Test Dataset\n",
 "df = pd.read_csv(\"KDDTest+.txt\")\n",
 "df_2, meta = arff.loadarff('KDDTest+.arff')\n",
```

```
 "df_2 = pd.DataFrame(df_2)"
]
},
{
"cell_type": "code",
"execution_count": 3,
"id": "7ed94bb1-44fd-47e1-aa33-a6b53ebc46c3",
"metadata": {
 "tags": []
},
"outputs": [
 {
  "data": {
   "text/html": [
    "<div>\n",
    "<style scoped>\n",
    "    .dataframe tbody tr th:only-of-type {\n",
    "        vertical-align: middle;\n",
    "    }\n",
    "\n",
    "    .dataframe tbody tr th {\n",
    "        vertical-align: top;\n",
    "    }\n",
    "\n",
    "    .dataframe thead th {\n",
```

```
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>0</th>\n",
"      <th>tcp</th>\n",
"      <th>ftp_data</th>\n",
"      <th>SF</th>\n",
"      <th>491</th>\n",
"      <th>0.1</th>\n",
"      <th>0.2</th>\n",
"      <th>0.3</th>\n",
"      <th>0.4</th>\n",
"      <th>0.5</th>\n",
"      <th>0.6</th>\n",
"      <th>0.7</th>\n",
"      <th>0.8</th>\n",
"      <th>0.9</th>\n",
"      <th>0.10</th>\n",
"      <th>0.11</th>\n",
"      <th>0.12</th>\n",
"      <th>0.13</th>\n",
```

```
"      <th>0.14</th>\n",
"      <th>0.15</th>\n",
"      <th>0.16</th>\n",
"      <th>0.18</th>\n",
"      <th>2</th>\n",
"      <th>2.1</th>\n",
"      <th>0.00</th>\n",
"      <th>0.00.1</th>\n",
"      <th>0.00.2</th>\n",
"      <th>0.00.3</th>\n",
"      <th>1.00</th>\n",
"      <th>0.00.4</th>\n",
"      <th>0.00.5</th>\n",
"      <th>150</th>\n",
"      <th>25</th>\n",
"      <th>0.17</th>\n",
"      <th>0.03</th>\n",
"      <th>0.17.1</th>\n",
"      <th>0.00.6</th>\n",
"      <th>0.00.7</th>\n",
"      <th>0.00.8</th>\n",
"      <th>0.05</th>\n",
"      <th>0.00.9</th>\n",
"      <th>normal</th>\n",
"      <th>20</th>\n",
```

```
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>0</td>\n",
"      <td>udp</td>\n",
"      <td>other</td>\n",
"      <td>SF</td>\n",
"      <td>146</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
```

```
"     <td>0</td>\n",
"     <td>0</td>\n",
"     <td>13</td>\n",
"     <td>1</td>\n",
"     <td>0.0</td>\n",
"     <td>0.0</td>\n",
"     <td>0.0</td>\n",
"     <td>0.0</td>\n",
"     <td>0.08</td>\n",
"     <td>0.15</td>\n",
"     <td>0.00</td>\n",
"     <td>255</td>\n",
"     <td>1</td>\n",
"     <td>0.00</td>\n",
"     <td>0.60</td>\n",
"     <td>0.88</td>\n",
"     <td>0.00</td>\n",
"     <td>0.00</td>\n",
"     <td>0.00</td>\n",
"     <td>0.00</td>\n",
"     <td>0.00</td>\n",
"     <td>normal</td>\n",
"     <td>15</td>\n",
"   </tr>\n",
"   <tr>\n",
```

```
    "      <th>1</th>\n",
    "      <td>0</td>\n",
    "      <td>tcp</td>\n",
    "      <td>private</td>\n",
    "      <td>S0</td>\n",
    "      <td>0</td>\n",
    "      <td>0</td>\n",
    "      <td>0</td>\n",
"      <td>77</td>\n",
    "      <td>0.30</td>\n",
    "      <td>0.03</td>\n",
    "      <td>0.30</td>\n",
    "      <td>0.00</td>\n",
    "      <td>0.00</td>\n",
    "      <td>0.00</td>\n",
    "      <td>0.00</td>\n",
    "      <td>0.00</td>\n",
    "      <td>normal</td>\n",
    "      <td>21</td>\n",
    "    </tr>\n",
    "  </tbody>\n",
    "</table>\n",
    "<p>125972 rows × 43 columns</p>\n",
    "</div>"
   ],
```

```
"text/plain": [
"       0  tcp  ftp_data   SF   491   0.1  0.2  0.3  0.4  0.5  0.6  0.7  0.8  \\\n",
"0      0  udp    other   SF   146    0    0    0    0    0    0    0    0  \n",
"1      0  tcp   private   S0    0    0    0    0    0    0    0    0    0  \n",
"2      0  tcp     http   SF   232  8153    0    0    0    0    0    1    0  \n",
"3      0  tcp     http   SF   199   420    0    0    0    0    0    1    0  \n",
"4      0  tcp   private  REJ    0    0    0    0    0    0    0    0    0  \n",
"...    ..  ...      ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  \n",
"125967  0  tcp   private   S0    0    0    0    0    0    0    0    0    0  \n",
"125968  8  udp   private   SF   105   145    0    0    0    0    0    0    0  \n",
"125969  0  tcp     smtp   SF  2231   384    0    0    0    0    0    1    0  \n",
"125970  0  tcp   klogin   S0    0    0    0    0    0    0    0    0    0  \n",
"125971  0  tcp  ftp_data   SF   151    0    0    0    0    0    0    1    0  \n",
"\n",
"       0.9  0.10  0.11  0.12  0.13  0.14  0.15  0.16  0.18    2  2.1  0.00  \\\n",
"0        0    0    0    0    0    0    0    0    0   13    1   0.0   \n",
"1        0    0    0    0    0    0    0    0    0  123    6   1.0   \n",
"2        0    0    0    0    0    0    0    0    0    5    5   0.2   \n",
"3        0    0    0    0    0    0    0    0    0   30   32   0.0   \n",
"4        0    0    0    0    0    0    0    0    0  121   19   0.0   \n",
"...    ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...   \n",
"125967    0    0    0    0    0    0    0    0    0  184   25   1.0   \n",
"125968    0    0    0    0    0    0    0    0    0    2    2   0.0   \n",
"125969    0    0    0    0    0    0    0    0    0    1    1   0.0   \n",
"125970    0    0    0    0    0    0    0    0    0  144    8   1.0   \n",
```

```
"125971   0    0    0    0    0    0    0    0    0    1    1   0.0  \n",
"\n",
"       0.00.1  0.00.2  0.00.3  1.00  0.00.4  0.00.5  150   25  0.17  0.03  \\\n",
"0        0.0    0.0    0.0  0.08    0.15    0.00  255    1  0.00  0.60   \n",
"1        1.0    0.0    0.0  0.05    0.07    0.00  255   26  0.10  0.05   \n",
"2        0.2    0.0    0.0  1.00    0.00    0.00   30  255  1.00  0.00   \n",
"3        0.0    0.0    0.0  1.00    0.00    0.09  255  255  1.00  0.00   \n",
"4        0.0    1.0    1.0  0.16    0.06    0.00  255   19  0.07  0.07   \n",
"...      ...    ...    ...   ...     ...     ...  ...  ...   ...   ...  \n",
"125967    1.0    0.0    0.0  0.14    0.06    0.00  255   25  0.10  0.06   \n",
"125968    0.0    0.0    0.0  1.00    0.00    0.00  255  244  0.96  0.01   \n",
"125969    0.0    0.0    0.0  1.00    0.00    0.00  255   30  0.12  0.06   \n",
"125970    1.0    0.0    0.0  0.06    0.05    0.00  255    8  0.03  0.05   \n",
"125971    0.0    0.0    0.0  1.00    0.00    0.00  255   77  0.30  0.03   \n",
"\n",
"       0.17.1  0.00.6  0.00.7  0.00.8  0.05  0.00.9   normal  20  \n",
"0        0.88    0.00    0.00    0.00  0.00    0.00   normal  15  \n",
"1        0.00    0.00    1.00    1.00  0.00    0.00  neptune  19  \n",
"2        0.03    0.04    0.03    0.01  0.00    0.01   normal  21  \n",
"3        0.00    0.00    0.00    0.00  0.00    0.00   normal  21  \n",
"4        0.00    0.00    0.00    0.00  1.00    1.00  neptune  21  \n",
"...      ...     ...     ...     ...   ...     ...      ... ..  \n",
"125967   0.00    0.00    1.00    1.00  0.00    0.00  neptune  20  \n",
"125968   0.01    0.00    0.00    0.00  0.00    0.00   normal  21  \n",
"125969   0.00    0.00    0.72    0.00  0.01    0.00   normal  18  \n",
```

```
    "125970    0.00    0.00    1.00    1.00  0.00    0.00  neptune  20  \n",
    "125971    0.30    0.00    0.00    0.00  0.00    0.00   normal  21  \n",
    "\n",
    "[125972 rows x 43 columns]"
   ]
  },
  "execution_count": 3,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "data"
]
},
{
"cell_type": "code",
"execution_count": 4,
"id": "8942e3f7-cee7-457e-8e55-42cee306b721",
"metadata": {
 "tags": []
},
"outputs": [
 {
  "data": {
```

```
"text/html": [
 "<div>\n",
 "<style scoped>\n",
 "    .dataframe tbody tr th:only-of-type {\n",
 "        vertical-align: middle;\n",
 "    }\n",
 "\n",
 "    .dataframe tbody tr th {\n",
 "        vertical-align: top;\n",
 "    }\n",
 "\n",
 "    .dataframe thead th {\n",
 "        text-align: right;\n",
 "    }\n",
 "</style>\n",
 "<table border=\"1\" class=\"dataframe\">\n",
 "  <thead>\n",
 "    <tr style=\"text-align: right;\">\n",
 "      <th></th>\n",
 "      <th>0</th>\n",
 "      <th>tcp</th>\n",
 "      <th>private</th>\n",
 "      <th>REJ</th>\n",
 "      <th>0.1</th>\n",
 "      <th>0.2</th>\n",
```

```
"      <th>0.3</th>\n",
"      <th>0.4</th>\n",
"      <th>0.5</th>\n",
"      <th>0.6</th>\n",
"      <th>0.7</th>\n",
"      <th>0.8</th>\n",
"      <th>0.9</th>\n",
"      <th>0.10</th>\n",
"      <th>0.11</th>\n",
"      <th>0.12</th>\n",
"      <th>0.13</th>\n",
"      <th>0.14</th>\n",
"    <td>0.00</td>\n",
"      <td>0.00</td>\n",
"      <td>0.00</td>\n",
"      <td>0.0</td>\n",
"      <td>0.44</td>\n",
"      <td>1.00</td>\n",
"      <td>mscan</td>\n",
"      <td>14</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"<p>22543 rows × 43 columns</p>\n",
"</div>"
```

```
],
"text/plain": [
"      0  tcp  private  REJ   0.1   0.2  0.3  0.4  0.5  0.6  0.7  0.8  \\\n",
"0     0  tcp  private  REJ    0     0   0   0   0   0   0   0  \n",
"1     2  tcp  ftp_data  SF  12983    0   0   0   0   0   0   0  \n",
"2     0  icmp  eco_i  SF   20     0   0   0   0   0   0   0  \n",
"3     1  tcp  telnet  RSTO   0    15   0   0   0   0   0   0  \n",
"4     0  tcp   http  SF   267  14515   0   0   0   0   0   1  \n",
"...  ..  ...     ...  ...  ...  ... ... ... ... ... ... ...  \n",
"22538  0  tcp   smtp  SF   794   333   0   0   0   0   0   1  \n",
"22539  0  tcp   http  SF   317   938   0   0   0   0   0   1  \n",
"22540  0  tcp   http  SF  54540  8314   0   0   0   2   0   1  \n",
"22541  0  udp  domain_u  SF   42    42   0   0   0   0   0   0  \n",
"22542  0  tcp  sunrpc  REJ    0     0   0   0   0   0   0   0  \n",
"\n",
"      0.9  0.10  0.11  0.12  0.13  0.14  0.15  0.16  0.17  0.18  229  10  \\\n",
"0       0    0    0    0    0    0    0    0    0    0  136   1  \n",
"1       0    0    0    0    0    0    0    0    0    0   1   1  \n",
"2       0    0    0    0    0    0    0    0    0    0   1  65  \n",
"3       0    0    0    0    0    0    0    0    0    0   1   8  \n",
"4       0    0    0    0    0    0    0    0    0    0   4   4  \n",
"...    ...   ...  ...  ...  ...  ...  ...  ...  ...  ...  ... ..  \n",
"22538   0    0    0    0    0    0    0    0    0    0   1   1  \n",
"22539   0    0    0    0    0    0    0    0    0    0   2  11  \n",
"22540   1    0    0    0    0    0    0    0    0    0   5  10  \n",
```

```
"22541   0    0    0    0    0    0    0    0    0    0   4   6  \n",
"22542   0    0    0    0    0    0    0    0    0    0   4  10  \n",
"\n",
"      0.00  0.00.1  1.00  1.00.1  0.04  0.06  0.00.2  255  10.1  0.04.1  \\\n",
"0      0.0   0.00   1.0    1.0  0.01  0.06   0.00  255    1   0.00  \n",
"1      0.0   0.00   0.0    0.0  1.00  0.00   0.00  134   86   0.61  \n",
"2      0.0   0.00   0.0    0.0  1.00  0.00   1.00   3   57   1.00  \n",
"3      0.0   0.12   1.0    0.5  1.00  0.00   0.75   29   86   0.31  \n",
"4      0.0   0.00   0.0    0.0  1.00  0.00   0.00  155  255   1.00  \n",
"...    ...   ...   ...   ...   ...   ...   ...   ...  ...   ...   ...  \n",
"22538  0.0   0.00   0.0    0.0  1.00  0.00   0.00  100  141   0.72  \n",
"22539  0.0   0.00   0.0    0.0  1.00  0.00   0.18  197  255   1.00  \n",
"22540  0.0   0.00   0.0    0.0  1.00  0.00   0.20  255  255   1.00  \n",
"22541  0.0   0.00   0.0    0.0  1.00  0.00   0.33  255  252   0.99  \n",
"22542  0.0   0.00   1.0    1.0  0.25  1.00   1.00  255   21   0.08  \n",
"\n",
"      0.06.1  0.00.3  0.00.4  0.00.5  0.00.6  1.00.2  1.00.3  neptune  21  \n",
"0       0.06    0.00    0.00    0.00     0.0    1.00    1.00  neptune  21  \n",
"1       0.04    0.61    0.02    0.00     0.0    0.00    0.00   normal  21  \n",
"2       0.00    1.00    0.28    0.00     0.0    0.00    0.00    saint  15  \n",
"3       0.17    0.03    0.02    0.00     0.0    0.83    0.71    mscan  11  \n",
"4       0.00    0.01    0.03    0.01     0.0    0.00    0.00   normal  21  \n",
"...      ...     ...     ...     ...     ...     ...     ...     ... ..  \n",
"22538   0.06    0.01    0.01    0.01     0.0    0.00    0.00   normal  21  \n",
"22539   0.00    0.01    0.01    0.01     0.0    0.00    0.00   normal  21  \n",
```

```
    "22540    0.00    0.00    0.00    0.00    0.0    0.07    0.07    back  15  \n",
    "22541    0.01    0.00    0.00    0.00    0.0    0.00    0.00   normal  21  \n",
    "22542    0.03    0.00    0.00    0.00    0.0    0.44    1.00    mscan  14  \n",
    "\n",
    "[22543 rows x 43 columns]"
   ]
  },
  "execution_count": 4,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "df"
]
},
{
"cell_type": "code",
"execution_count": 5,
"id": "c2df8e8f-2d2e-4420-9958-e98e1b8d7965",
"metadata": {
 "tags": []
},
"outputs": [
 {
```

```
"data": {
 "text/html": [
  "<div>\n",
  "<style scoped>\n",
  "    .dataframe tbody tr th:only-of-type {\n",
  "        vertical-align: middle;\n",
  "    }\n",
  "\n",
  "    .dataframe tbody tr th {\n",
  "        vertical-align: top;\n",
  "    }\n",
  "\n",
  "    .dataframe thead th {\n",
  "        text-align: right;\n",
  "    }\n",
  "</style>\n",
  "<table border=\"1\" class=\"dataframe\">\n",
  "  <thead>\n",
  "    <tr style=\"text-align: right;\">\n",
  "      <th></th>\n",
  "      <th>duration</th>\n",
  "      <th>protocol_type</th>\n",
  "      <th>service</th>\n",
  "      <th>flag</th>\n",
  "      <th>src_bytes</th>\n",
```

```
"      <th>dst_bytes</th>\n",
"      <th>land</th>\n",
"      <th>wrong_fragment</th>\n",
"      <th>urgent</th>\n",
"      <th>hot</th>\n",
"      <th>num_failed_logins</th>\n",
"    <td>0.30</td>\n",
"      <td>0.03</td>\n",
"      <td>0.30</td>\n",
"      <td>0.00</td>\n",
"      <td>0.00</td>\n",
"      <td>0.00</td>\n",
"      <td>0.00</td>\n",
"      <td>0.00</td>\n",
"      <td>b'normal'</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"<p>125973 rows × 42 columns</p>\n",
"</div>"
],
"text/plain": [
"       duration protocol_type      service   flag  src_bytes  dst_bytes  \\\n",
"0           0.0        b'tcp'  b'ftp_data'  b'SF'      491.0        0.0   \n",
"1           0.0        b'udp'     b'other'  b'SF'      146.0        0.0   \n",
```

```
"2          0.0       b'tcp'   b'private'  b'S0'       0.0       0.0   \n",
"3          0.0       b'tcp'     b'http'  b'SF'     232.0     8153.0   \n",
"4          0.0       b'tcp'     b'http'  b'SF'     199.0      420.0   \n",
"...        ...         ...         ...    ...       ...        ...   \n",
"125968     0.0       b'tcp'   b'private'  b'S0'       0.0       0.0   \n",
"125969     8.0       b'udp'   b'private'  b'SF'     105.0     145.0   \n",
"125970     0.0       b'tcp'     b'smtp'  b'SF'    2231.0     384.0   \n",
"125971     0.0       b'tcp'    b'klogin'  b'S0'       0.0       0.0   \n",
"125972     0.0       b'tcp'  b'ftp_data'  b'SF'     151.0       0.0   \n",
"\n",
"       land  wrong_fragment  urgent  hot  num_failed_logins logged_in  \\\n",
"0      b'0'           0.0     0.0  0.0              0.0      b'0'   \n",
"1      b'0'           0.0     0.0  0.0              0.0      b'0'   \n",
"2      b'0'           0.0     0.0  0.0              0.0      b'0'   \n",
"3      b'0'           0.0     0.0  0.0              0.0      b'1'   \n",
"4      b'0'           0.0     0.0  0.0              0.0      b'1'   \n",
"...     ...           ...     ...  ...              ...       ...   \n",
"125968  b'0'          0.0     0.0  0.0              0.0      b'0'   \n",
"125969  b'0'          0.0     0.0  0.0              0.0      b'0'   \n",
"125970  b'0'          0.0     0.0  0.0              0.0      b'1'   \n",
"125971  b'0'          0.0     0.0  0.0              0.0      b'0'   \n",
"125972  b'0'          0.0     0.0  0.0              0.0      b'1'   \n",
"\n",
"       num_compromised  root_shell  su_attempted  num_root  \\\n",
"0              0.0         0.0          0.0       0.0   \n",
```

```
    "1              0.0       0.0       0.0      0.0  \n",
    "2              0.0       0.0       0.0      0.0  \n",
    "3              0.0       0.0       0.0      0.0  \n",
    "4              0.0       0.0       0.0      0.0  \n",
    "...            ...       ...       ...      ...  \n",
    "125968         0.0       0.0       0.0      0.0  \n",
    "125969         0.0       0.0       0.0      0.0  \n",
    "125970         0.0       0.0       0.0      0.0  \n",
    "125971         0.0       0.0       0.0      0.0  \n",
    "125972         0.0       0.0       0.0      0.0  \n",
    "\n",
    "        num_file_creations  num_shells  num_access_files  num_outbound_cmds
\\\n",
    "0               0.0       0.0           0.0              0.0  \n",
    "1               0.0       0.0           0.0              0.0  \n",
    "2               0.0       0.0           0.0              0.0  \n",
    "3               0.0       0.0           0.0              0.0  \n",
    "4               0.0       0.0           0.0              0.0  \n",
    "...             ...       ...           ...              ...  \n",
    "125968          0.0       0.0           0.0              0.0  \n",
    "125969          0.0       0.0           0.0              0.0  \n",
    "125970          0.0       0.0           0.0              0.0  \n",
    "125971          0.0       0.0           0.0              0.0  \n",
    "125972          0.0       0.0           0.0              0.0  \n",
    "\n",
```

```
"      is_host_login is_guest_login  count  srv_count  serror_rate  \\\n",
"0            b'0'         b'0'    2.0       2.0         0.0   \n",
"1            b'0'         b'0'   13.0       1.0         0.0   \n",
"2            b'0'         b'0'  123.0       6.0         1.0   \n",
"3            b'0'         b'0'    5.0       5.0         0.2   \n",
"4            b'0'         b'0'   30.0      32.0         0.0   \n",
"...           ...          ...    ...       ...         ...   \n",
"125968       b'0'         b'0'  184.0      25.0         1.0   \n",
"125969       b'0'         b'0'    2.0       2.0         0.0   \n",
"125970       b'0'         b'0'    1.0       1.0         0.0   \n",
"125971       b'0'         b'0'  144.0       8.0         1.0   \n",
"125972       b'0'         b'0'    1.0       1.0         0.0   \n",
"\n",
"      srv_serror_rate  rerror_rate  srv_rerror_rate  same_srv_rate  \\\n",
"0                 0.0          0.0              0.0           1.00   \n",
"1                 0.0          0.0              0.0           0.08   \n",
"2                 1.0          0.0              0.0           0.05   \n",
"3                 0.2          0.0              0.0           1.00   \n",
"4                 0.0          0.0              0.0           1.00   \n",
"...               ...          ...              ...           ...   \n",
"125968            1.0          0.0              0.0           0.14   \n",
"125969            0.0          0.0              0.0           1.00   \n",
"125970            0.0          0.0              0.0           1.00   \n",
"125971            1.0          0.0              0.0           0.06   \n",
"125972            0.0          0.0              0.0           1.00   \n",
```

```
"\n",
"       diff_srv_rate  srv_diff_host_rate  dst_host_count  dst_host_srv_count  \\\n",
"0              0.00               0.00          150.0              25.0   \n",
"1              0.15               0.00          255.0               1.0   \n",
"2              0.07               0.00          255.0              26.0   \n",
"3              0.00               0.00           30.0             255.0   \n",
"4              0.00               0.09          255.0             255.0   \n",
"...             ...                ...            ...               ...   \n",
"125968         0.06               0.00          255.0              25.0   \n",
"125969         0.00               0.00          255.0             244.0   \n",
"125970         0.00               0.00          255.0              30.0   \n",
"125971         0.05               0.00          255.0               8.0   \n",
"125972         0.00               0.00          255.0              77.0   \n",
"\n",
"       dst_host_same_srv_rate  dst_host_diff_srv_rate  \\\n",
"0                        0.17                    0.03   \n",
"1                        0.00                    0.60   \n",
"2                        0.10                    0.05   \n",
"3                        1.00                    0.00   \n",
"4                        1.00                    0.00   \n",
"...                       ...                     ...   \n",
"125968                   0.10                    0.06   \n",
"125969                   0.96                    0.01   \n",
"125970                   0.12                    0.06   \n",
"125971                   0.03                    0.05   \n",
```

```
"125972                0.30                0.03   \n",
"\n",
"       dst_host_same_src_port_rate  dst_host_srv_diff_host_rate  \\\n",
"0                         0.17                    0.00   \n",
"1                         0.88                    0.00   \n",
"2                         0.00                    0.00   \n",
"3                         0.03                    0.04   \n",
"4                         0.00                    0.00   \n",
"...                       ...                     ...   \n",
"125968                    0.00                    0.00   \n",
"125969                    0.01                    0.00   \n",
"125970                    0.00                    0.00   \n",
"125971                    0.00                    0.00   \n",
"125972                    0.30                    0.00   \n",
"\n",
"       dst_host_serror_rate  dst_host_srv_serror_rate  dst_host_rerror_rate  \\\n",
"0                      0.00                      0.00                  0.05   \n",
"1                      0.00                      0.00                  0.00   \n",
"2                      1.00                      1.00                  0.00   \n",
"3                      0.03                      0.01                  0.00   \n",
"4                      0.00                      0.00                  0.00   \n",
"...                    ...                       ...                   ...   \n",
"125968                 1.00                      1.00                  0.00   \n",
"125969                 0.00                      0.00                  0.00   \n",
"125970                 0.72                      0.00                  0.01   \n",
```

```
   "125971              1.00                 1.00              0.00   \n",
   "125972              0.00                 0.00              0.00   \n",
   "\n",
   "       dst_host_srv_rerror_rate       class  \n",
   "0                      0.00   b'normal'  \n",
   "1                      0.00   b'normal'  \n",
   "2                      0.00  b'anomaly'  \n",
   "3                      0.01   b'normal'  \n",
   "4                      0.00   b'normal'  \n",
   "...                     ...         ...  \n",
   "125968                    0.00  b'anomaly'  \n",
   "125969                    0.00   b'normal'  \n",
   "125970                    0.00   b'normal'  \n",
   "125971                    0.00  b'anomaly'  \n",
   "125972                    0.00   b'normal'  \n",
   "\n",
   "[125973 rows x 42 columns]"
  ]
 },
 "execution_count": 5,
 "metadata": {},
 "output_type": "execute_result"
}
],
"source": [
```

```
 "data_2"
]
},
{
"cell_type": "code",
"execution_count": 6,
"id": "ae5ed70f-f39c-40e6-9cd1-df95a7675995",
"metadata": {
 "tags": []
},
"outputs": [
 {
  "data": {
   "text/html": [
    "<div>\n",
    "<style scoped>\n",
    "    .dataframe tbody tr th:only-of-type {\n",
    "        vertical-align: middle;\n",
    "    }\n",
    "\n",
    "    .dataframe tbody tr th {\n",
    "        vertical-align: top;\n",
    "    }\n",
    "\n",
    "    .dataframe thead th {\n",
```

```
"       text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>duration</th>\n",
"      <th>protocol_type</th>\n",
"      <th>service</th>\n",
"      <th>flag</th>\n",
"      <th>src_bytes</th>\n",
"      <th>dst_bytes</th>\n",
"      <th>land</th>\n",
"      <th>wrong_fragment</th>\n",
"      <th>urgent</th>\n",
"      <th>hot</th>\n",
"      <th>num_failed_logins</th>\n",
"      <th>logged_in</th>\n",
"      <th>num_compromised</th>\n",
"      <th>root_shell</th>\n",
"      <th>su_attempted</th>\n",
"      <th>num_root</th>\n",
"      <th>num_file_creations</th>\n",
"      <th>num_shells</th>\n",
```

```
"    <th>num_access_files</th>\n",
"    <th>num_outbound_cmds</th>\n",
"    <th>is_host_login</th>\n",
"    <th>is_guest_login</th>\n",
"    <th>count</th>\n",
"    <th>srv_count</th>\n",
"    <th>serror_rate</th>\n",
"    <th>srv_serror_rate</th>\n",
"    <th>rerror_rate</th>\n",
"    <th>srv_rerror_rate</th>\n",
"    <th>same_srv_rate</th>\n",
"    <th>diff_srv_rate</th>\n",
"    <th>srv_diff_host_rate</th>\n",
"    <th>dst_host_count</th>\n",
"    <th>dst_host_srv_count</th>\n",
"    <th>dst_host_same_srv_rate</th>\n",
"    <th>dst_host_diff_srv_rate</th>\n",
"    <th>dst_host_same_src_port_rate</th>\n",
"    <th>dst_host_srv_diff_host_rate</th>\n",
"    <th>dst_host_serror_rate</th>\n",
"    <th>dst_host_srv_serror_rate</th>\n",
"    <th>dst_host_rerror_rate</th>\n",
"    <th>dst_host_srv_rerror_rate</th>\n",
"    <th>class</th>\n",
"  </tr>\n",
```

```
" </thead>\n",
" <tbody>\n",
"  <tr>\n",
"   <th>0</th>\n",
"   <td>0.0</td>\n",
"   <td>b'tcp'</td>\n",
"   <td>b'private'</td>\n",
"   <td>b'REJ'</td>\n",
"   <td>0.0</td>\n",
"   <td>0.0</td>\n",
"   <td>b'0'</td>\n",
"   <td>0.00</td>\n",
"   <td>0.00</td>\n",
"   <td>0.0</td>\n",
"   <td>0.44</td>\n",
"   <td>1.00</td>\n",
"   <td>b'anomaly'</td>\n",
"  </tr>\n",
" </tbody>\n",
"</table>\n",
"<p>22544 rows × 42 columns</p>\n",
"</div>"
],
"text/plain": [
"      duration protocol_type      service     flag  src_bytes  dst_bytes  \\\n",
```

```
"0         0.0       b'tcp'   b'private'   b'REJ'       0.0       0.0   \n",
"1         0.0       b'tcp'   b'private'   b'REJ'       0.0       0.0   \n",
"2         2.0       b'tcp'  b'ftp_data'    b'SF'    12983.0       0.0   \n",
"3         0.0      b'icmp'    b'eco_i'    b'SF'       20.0       0.0   \n",
"4         1.0       b'tcp'    b'telnet'  b'RSTO'       0.0      15.0   \n",
"...       ...         ...         ...      ...        ...       ...   \n",
"22539     0.0       b'tcp'     b'smtp'    b'SF'     794.0     333.0   \n",
"22540     0.0       b'tcp'     b'http'    b'SF'     317.0     938.0   \n",
"22541     0.0       b'tcp'     b'http'    b'SF'   54540.0    8314.0   \n",
"22542     0.0       b'udp'  b'domain_u'   b'SF'      42.0      42.0   \n",
"22543     0.0       b'tcp'    b'sunrpc'   b'REJ'       0.0       0.0   \n",
"\n",
"      land  wrong_fragment  urgent  hot  num_failed_logins logged_in  \\\n",
"0     b'0'           0.0     0.0  0.0             0.0      b'0'   \n",
"1     b'0'           0.0     0.0  0.0             0.0      b'0'   \n",
"2     b'0'           0.0     0.0  0.0             0.0      b'0'   \n",
"3     b'0'           0.0     0.0  0.0             0.0      b'0'   \n",
"4     b'0'           0.0     0.0  0.0             0.0      b'0'   \n",
"...    ...           ...     ...  ...             ...       ...   \n",
"22539 b'0'           0.0     0.0  0.0             0.0      b'1'   \n",
"22540 b'0'           0.0     0.0  0.0             0.0      b'1'   \n",
"22541 b'0'           0.0     0.0  2.0             0.0      b'1'   \n",
"22542 b'0'           0.0     0.0  0.0             0.0      b'0'   \n",
"22543 b'0'           0.0     0.0  0.0             0.0      b'0'   \n",
"\n",
```

```
"      num_compromised  root_shell  su_attempted  num_root \\\n",
"0              0.0       0.0         0.0       0.0   \n",
"1              0.0       0.0         0.0       0.0   \n",
"2              0.0       0.0         0.0       0.0   \n",
"3              0.0       0.0         0.0       0.0   \n",
"4              0.0       0.0         0.0       0.0   \n",
"...            ...       ...         ...       ...   \n",
"22539          0.0       0.0         0.0       0.0   \n",
"22540          0.0       0.0         0.0       0.0   \n",
"22541          1.0       0.0         0.0       0.0   \n",
"22542          0.0       0.0         0.0       0.0   \n",
"22543          0.0       0.0         0.0       0.0   \n",
"\n",
"       num_file_creations  num_shells  num_access_files  num_outbound_cmds
\\\n",
"0                 0.0        0.0             0.0               0.0   \n",
"1                 0.0        0.0             0.0               0.0   \n",
"2                 0.0        0.0             0.0               0.0   \n",
"3                 0.0        0.0             0.0               0.0   \n",
"4                 0.0        0.0             0.0               0.0   \n",
"...               ...        ...             ...               ...   \n",
"22539             0.0        0.0             0.0               0.0   \n",
"22540             0.0        0.0             0.0               0.0   \n",
"22541             0.0        0.0             0.0               0.0   \n",
"22542             0.0        0.0             0.0               0.0   \n",
```

```
"22543              0.0        0.0             0.0             0.0   \n",
"\n",
"      is_host_login is_guest_login  count  srv_count  serror_rate  \\\n",
"0            b'0'          b'0'  229.0      10.0         0.0   \n",
"1            b'0'          b'0'  136.0       1.0         0.0   \n",
"2            b'0'          b'0'    1.0       1.0         0.0   \n",
"3            b'0'          b'0'    1.0      65.0         0.0   \n",
"4            b'0'          b'0'    1.0       8.0         0.0   \n",
"...           ...           ...    ...       ...         ...   \n",
"22539        b'0'          b'0'    1.0       1.0         0.0   \n",
"22540        b'0'          b'0'    2.0      11.0         0.0   \n",
"22541        b'0'          b'0'    5.0      10.0         0.0   \n",
"22542        b'0'          b'0'    4.0       6.0         0.0   \n",
"22543        b'0'          b'0'    4.0      10.0         0.0   \n",
"\n",
"      srv_serror_rate  rerror_rate  srv_rerror_rate  same_srv_rate  \\\n",
"0              0.00         1.0             1.0           0.04   \n",
"1              0.00         1.0             1.0           0.01   \n",
"2              0.00         0.0             0.0           1.00   \n",
"3              0.00         0.0             0.0           1.00   \n",
"4              0.12         1.0             0.5           1.00   \n",
"...             ...         ...             ...           ...   \n",
"22539          0.00         0.0             0.0           1.00   \n",
"22540          0.00         0.0             0.0           1.00   \n",
"22541          0.00         0.0             0.0           1.00   \n",
```

```
"22542          0.00        0.0          0.0         1.00   \n",
"22543          0.00        1.0          1.0         0.25   \n",
"\n",
"      diff_srv_rate  srv_diff_host_rate  dst_host_count  dst_host_srv_count  \\\n",
"0            0.06            0.00         255.0             10.0   \n",
"1            0.06            0.00         255.0              1.0   \n",
"2            0.00            0.00         134.0             86.0   \n",
"3            0.00            1.00           3.0             57.0   \n",
"4            0.00            0.75          29.0             86.0   \n",
"...           ...             ...           ...              ...   \n",
"22539        0.00            0.00         100.0            141.0   \n",
"22540        0.00            0.18         197.0            255.0   \n",
"22541        0.00            0.20         255.0            255.0   \n",
"22542        0.00            0.33         255.0            252.0   \n",
"22543        1.00            1.00         255.0             21.0   \n",
"\n",
"      dst_host_same_srv_rate  dst_host_diff_srv_rate  \\\n",
"0                       0.04                    0.06   \n",
"1                       0.00                    0.06   \n",
"2                       0.61                    0.04   \n",
"3                       1.00                    0.00   \n",
"4                       0.31                    0.17   \n",
"...                      ...                     ...   \n",
"22539                   0.72                    0.06   \n",
"22540                   1.00                    0.00   \n",
```

"22541 1.00 0.00 \n",

"22542 0.99 0.01 \n",

"22543 0.08 0.03 \n",

"\n",

" dst_host_same_src_port_rate dst_host_srv_diff_host_rate \\\n",

"0 0.00 0.00 \n",

"1 0.00 0.00 \n",

"2 0.61 0.02 \n",

"3 1.00 0.28 \n",

"4 0.03 0.02 \n",

"... ... ... \n",

"22539 0.01 0.01 \n",

"22540 0.01 0.01 \n",

"22541 0.00 0.00 \n",

"22542 0.00 0.00 \n",

"22543 0.00 0.00 \n",

"\n",

" dst_host_serror_rate dst_host_srv_serror_rate dst_host_rerror_rate \\\n",

"0 0.00 0.0 1.00 \n",

"1 0.00 0.0 1.00 \n",

"2 0.00 0.0 0.00 \n",

"3 0.00 0.0 0.00 \n",

"4 0.00 0.0 0.83 \n",

"... ... ... \n",

"22539 0.01 0.0 0.00 \n",

```
 "22540              0.01                0.0             0.00   \n",
 "22541              0.00                0.0             0.07   \n",
 "22542              0.00                0.0             0.00   \n",
 "22543              0.00                0.0             0.44   \n",
 "\n",
 "      dst_host_srv_rerror_rate       class  \n",
 "0                     1.00  b'anomaly'  \n",
 "1                     1.00  b'anomaly'  \n",
 "2                     0.00   b'normal'  \n",
 "3                     0.00  b'anomaly'  \n",
 "4                     0.71  b'anomaly'  \n",
 "...                    ...         ...  \n",
 "22539                   0.00   b'normal'  \n",
 "22540                   0.00   b'normal'  \n",
 "22541                   0.07  b'anomaly'  \n",
 "22542                   0.00   b'normal'  \n",
 "22543                   1.00  b'anomaly'  \n",
 "\n",
 "[22544 rows x 42 columns]"
 ]
},
"execution_count": 6,
"metadata": {},
"output_type": "execute_result"
}
```

],
"source": [
"df_2"
]
},
{
"cell_type": "code",
"execution_count": 7,
"id": "316c5da3-adf4-4043-af99-303880d54541",
"metadata": {
"tags": []
},
"outputs": [
{
"data": {
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAAjoAAAIjCAYAAAAKkbGTAAAAOXRF
WHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlcnNpb24zLjcuMiwgaHR0cHM6Ly9tY
XRwbG90bGliLm9yZy8pXeV/AAAACXBIWXMAAA9hAAAPYQGoP6dpAACQf
ElEQVR4nOzdeVxU1f8/8NeAgIAwsiOJghuCYCmWIimYiuZGWmmhKC6ouaKi6a
fcyi33UnNPNBcq99IQVxR3UTTcd3DBFcEFAeH8/vDH/ToMInMZVK6v5+Mxj+L
OmfecGWd5z7nnvI9KCCFAREREpEAGb7oDREMWFiQ4REREpFhMdIiIiUi
wmOkRERKRYTHSIiIhIsZjoEBERkWIx0SEiIiLFYqJDREREisVEh4iIBSLiQ69ky
IiIqBSqaRL6dKl4ejoiEaNGmHixIm4ffu21m3GjBkDlUqlcSwzMxO09e/dGuXXLlYGh
oiA8++AAAcP/+fXz11Vewt7eHSqXCZ5999hoelTybN2/GmDFj9B533Dt+GDNmD

B48eKB13a+//oqIiAi932dh5f77Hzly5LXeb0hICFxcXDSOqVQqnZ9/uf9mee+rOJ6H
GzduYMyYMYiPj9e6Lr/3EFFxY6JD77QlS5Zg//792Lp1K+bMmYMPPvgAP/30E9z
d3bFt2zaNtj169MD+/fs1js2dOxfz58/Hd999h9jYWPz+++8AgB9//BHr1q3DjBkzsH//
fkyePPm1PSZdbd68GWPHjtV73H379mHs2LFvZaLzNtm/fz969Oih023k/pvJuS9d3b
hxA2PHjs030cnvPURU3Eq96Q4QvUmenp6oU6eO9Pfnn3+n2nfnn3+OQYMG4ePP0a7du1
w/vx5ODg4AADKly+P8uXLa9w+ISEBpqam6Nevn9PHjnrra3p6OkxNTf
UWT2mEEHj69GmJe47q1atXXrPFfffF6K+75eJb/3EFFx44gOUR4VKlTAtGnT8PDh
Q8yfP186nnfYXaVSYdGiRUhPT5dOgeWeCti2bRtOnz4tHd+1axeA56e6xo0bh+rVq
8PExAR2dnbo2rUr7ty5o9EHFxcXtGrVCmvXrkWtWrVQunRp6eKdhcqevXdTePXjynHd1XmsqHmh3Pnw5XV1eUKVMGPj4+OHDgg
NQuJCQEc+bMkR5P7uXKlSsvfX62btKwMBAlC9fHqVL0aVKlXQq1cv3L17V+
O5Gjp0KADA1dVV43lwcXHByZMnERMTI3PPZ3z9lTDBkyBB988AHUajWsr
a3h4+ODDRs2aPVDpVKhX79+mDdvHtzd3WFiYoKlS5S5cCAM6cOYOvv/4aDg4O
MDExQYUKFdC5c2dkZGS89HHdvHkT3t7eqFq1Ks6fPw8AuHTpEr766is4OTnBx
MQEDg4OaNy4cb6jFXlFRETAx8dHcGBHd3x7Jly/Jtl/d00pMnTxAeHg5XV1eUL
l0a1tbWqFOnDlatWgXg1f9mBT0vLztNlpKSgq5du26pNHGxc
UFISEhWrf19/eHv78/AGDXrl348MMPAQBdu3aV+pZ7n/dFRaF27dowwN
TVF9erV8dtvv2m0e9XzQvQyHNEhKeLFi3Z3bt3v7TN/v37YWlpidGTJ3n2jnv7ZxbKa7x6K
MHgOdf6vv370efPn2QmpqKFStYW7RkvRpGrxKxPW+5a0eywNHBa0e9s8KKHKIm8cTqLZaQV5d6WeFcEA60Ib4P6f3PZLQM
MHgOdf6vv370efPn2QmpqKFStWAAA8PDyQk5ODwMBA7NmzB8OGDUP9+vV
x9epVjB49Gv7+/jhy5IjGaMTRo0dx+vRpfP/993B1dYW3t7eFtt2/dHdjUTB49kK+N6fjLDiYKs
NGjULlypWxf/9+jBs3dl8uXMGSJUs0+hARUYnpx5Lq0nR8d3ETSyXiY0B8ZRRHrrt+vJlY7thtZa8
Ll6FWqzFy5Eg8fvwYKl+1jilUK5cuZc+7osXL8LHxwc9evSAWq3GlStXMH36dBn0dxHxcEh36X2SU
z88cf477//YGRkhB49euD+3DiX/Yo2p8uN3c8/KXWqmLV1JU7fs8pEt/3MAHwq6oK3xDe7d5DK3n
FAJiYmAAAMjIycP/+fYSHh2PUgiedHk2wtvlSr4Ppw/RvLfRqQ6eKRfzS9tgAOa36txyXnmh8
Xv2YNSoUXB0dIS9vT2uLwV8OFwLu92/ouvkuxXrNO+ZLnNq4l02Gc+Yb2m

KSEhAixYtUL58eezfvx+2trbS6yA7OxuTJ09GhQoVcPfuXezbty/f03EvioiIQNeuXR
EYGIhp06YhNTUVY8aMQUZGBgwMCv59OXjwYPz+++8YN24catWqhcePHyM
hIQH37t2T/g1f9W+W3/NSkO7du6Np06ZYuXIlkpKS8P3338Pf3x8nTpxA2bJlC7zti
2rXro0lS5aga9eu+P7779GyZUsAeOUozvHjxzFkyBAMHz4cDg4OWLRoEbp3744q
VaqgYcOGhXpeiF5KEL2DlixZIgCIw4cPv7SNg4ODcHd3l/4ePXq0yPuW6dKlizA3
N9e6rZ+fn6hRo4bGsVWrVgkAYs2aNRrHDx8+LACIX3/9VTpWsWJFYWhoKM6
ePavRtlevXqJMmTLi6tWrGsenTp0qAIiTJ08KIYS4fPmyACC8vLzEs2fPpHaHDh0S
AMSqVaukY3379tV6XIWVk8ejfv36UKlUiI2NRUREBH7++WdWdZMfW9wW+usm
XL4ubNm1qnyY4dOyZ7YUrZsmW1vkuKWvMqL87RKUCZMmUQFxcHNzc3vcY
tSUuh09LS0KJFC5w8eRIPHz6EpOT4ePjg82bN8tacXDjxg00atQIhoaGOH/+
POrUqSNVGd29e7esmkJ5V1wIIXDz5k2MGTMGZ86cQXR0NCIiIvDzzz+jYcOGWLx4
W3edcFy9exLhx44pU7wZ4+by6Y8eOwc/PT6cVPydOnADDwfLfpHTt2aCS22dnZiIq
Kwvz582U/r7mr6/J+MYsiTkYOCgpCnTp1MHjwYIwfPx5ubm5YsXIlAMDsXXrVtSuXb
vQk5ELW48HgOyiiXknsGZlZeHYsWMYOXIkxo8fj+PHj+P48eMYPHgwnJ2dsXjxYgDA
Vmzz1HKtW7cO06ZN06ZN06ZN06ZN06ZN06ZN06ZN06ZN06ZN06ZN06ZN06ZN06ZN
LduU3ULnzp3RuXLnzp3RuXJypO+SK1euYP369Vi/fj0uXLiAiIgIrF+/HhcvXsSV
Eb369Xj99dfx+uuvo1+/fnjttdfQr18/vPbaa3j99dfx+uuvo1+/fnjttdfQr18/vPba
a3j99dfx+uuvo1+/fnjttdfQr18/vPba63j99dfx+uuvo1+/fnjttdfQr18/vPba63j
P78+ahWrZo+uwi1Wo2jR4+icuXKjR4+icuXKGonO1atX4ebmtX4ebmhqdPnxYp/oULF3Dq1Ck
8q5E08X9+/elSci7du3CyMnYW1tjYYNG6JRo0bqJRiBgYF48OABABVq1aJSW
N169fR8eOHWlZaXTJOcXX6v5ffyamppi1qxZ6Natm6Natm6y+vmpSbmEn4uZ1//59PH3
6FE5OTsjJycHcXBwcHBzQ1eHkZER7t69ER7t69i+nTpxdpB4JcTHQKcPHiRfTu
5JFajllIUICAhAu3bt0Lt3bzk4ADVq1eHkZER7t69ER7t69i+nTpxdpB4JcTHQKcPHiRfTu

3RudOnWCp6en1hCwrl9GuYpjKbSBgQGSk5O1RkNu3bqFChUqyF5W+6KnT5/Cx
MREb6fw9Cnvh7uBgQHs7OxQpUoVrWqmRfHo0SOtuiRyTgXcuXMHXbt2xb///pv
v9XJ+ddvZ2WHfvn163+/LwcEBUVFRqFWrlkaiEx0dje7du0srhuRYvHgxZsyYIa20
q1q1KsLCwtCjRw9Z8QwNDWFra4sGDRpIp6v0Uao+KSkJgYGBSEhIgLOzs3QK1
8vLCxs2bNCpkuvVq1chhEClSpVw6NAh2NnZSdcZGxvD3t5edh2hkqZbt274+eefY
WFhoXH88ePH6N+/P3777Te7Te93t/p06fx4Ycf6lzstXTp0jh9+nShVzbJoc+6R/3794eRk
ZHWqb/w8HCkp6djzpw5RerrpUuXcOTIEahUKtSqVatIP05sbW0RExODGjVqYNG
iRZg1axaOHTuGNWvWYNSoUYzEdVYQztEpwJ07d3Dx4kWNMuUqlarIQ8AuLi
56Wwr94oZvW7Zs0Rjuz87Oxvbt24tUZTcnJwfjx4/HvHnzcOvWLanmzciRI+Hi4iJ7
Gezvv/+O+fPn49KlS9KlS9JmhjNmzEClSpVkDdfXr2idI3yKM6EJDk5GXPnztWYU9a3b
19pYmZJI2dPouzsbKxfv16jEGFgYKDsCuElaWuNd13uv1XelaK3bt2Cs7MzMjMzZ
cfu1q0bfH19WRiwGPHU1RvQo0cPrFy5EiNHjnzTXXmly5cv6z3m5MmTMXToU
MydOxeenp56/+uKlOmDOLj4zF//nwwYGhri8ePHaNeuHrjxo3S/2/Zsk
ID58+crqhy9rr/vLly4gBYtWuD69etk5OdsiStLXGu2rjxo3S/2/Zsk
WjAnJ2dja2b99e6B+DLzN79mx8+eWX2LNnj5Os5E2jui8AcVViFBfBg8ejB9//BH
m5uYPHjwS9upVCpZS5ZfrB9ibGystVcQ64fIU1z1iYp7h+k3QdfNF1u0aAEhBFa
sWAFra2sz/XTp06wcDAAJs2bdK5D/mNEqxZswZdunRBeno6Zs2aZdSwZdunRBeno6R3R0VNBnVV6F
+Yw9fvy4Vj2mFxkZGcHFxQXTpk1Dtk1Dq1atCn3feS1atAi9e/eGqakpAgICMGbMGDRo0ACm
to4ovMGnDhxAh988AGA988AGA53u9vOhtmJh87NgxaQSgoB2v5fb1bSuIqBQv+83y6NEjl
C5dWnbczp07Y/HixXYzp2DoaFhofc9q127tpSIurq64v
VOPb555/Dzc0NcXFxRervuyjvZ1VcXByys7OlCtnnzp2DoaFhofc9q127tpSIurq64v
Dhw1r/Xvrw/fff44cffsDw4cNhYGCAqCg9/jEROeN2Llz55vuQoFe7F9x9JWF0PQr95esS
qXCqFGjNFZeZWdn4+DBg1JiLUdx7TBdkpiYmORbffRo0cwNjaWFbNixYoAtCt
Z16hRg6d0ZXjxs2r69OmwsLA0qVLpaJ7KSkp6Nq1Kxo0aFCoe6Nq1Kxo0aFCoeGXLlsXly5dx5dhb2+
PxMREnU93FlZmZiY6dOjAJKcY8dQVvRF5J3Z6eHigTZs2sid2vstyl3jHxMTAx8d
PxMREnU93FlZmZiY6dOjAJKcY8dQVvRF5J3Z6eHigTZs2sid2vstyl3jHxMTAx8d

H44vX2NgYLi4uCA8Pl33qqbi2QnmTdD111blzZxw9ehSLFy+Wqi4fPHgQoaGh8P
b2RkREhM59KElba5Q07733HqKjo1GjRg2N4wkJCQgICChUodOePXti6dKlcHJyQ
mJiIsqXL//Sz6einF4aNGgQ7Ozs8L///U92DCoYR3TotSuOiZ3vstxfsl27dsXPP/+s92
XQb/sIpBwNGjTQmhtWkF9++QVdunSBj4+PNKfu2bNnaNOmDX7++WdZfRg0aB
BKlSqFxMREuLu7S8c7dOiAQYMGMdEpgrS0NNy6dUsr0bl9+3a+I3P5WbBgAdq
1a4cLFy5gwIABCA0NLZbaX9nZ2Zg8eTK2bNnyVs7ZVAKO6NBrVxwTO4mA5+
X3jYyMpM0XN2zYgCVLLsDDwwNjxoyRfZop14ULF3D69GkIIeDh4YEqVarIjlW
SttYoaTp37oyYmBhMmzZNKmNx4MABtRRr1q1ite1a1f88ssvxZLoK
HHE9G3DRIdeO3Nzcxw4cEBrJ+Djx4/D19eXH/BvocOHD+Ovv/5CYmKiVs2QtW
vXvqFeafvfwwww8xfPhwfP7557h06RJq1KiBtm3b4vDhw2jZsmWxT4TXpT6PhYUF
h49iqpVq2okOocPH0bz5s1x7969Yu2rkj158gPTh4eH47bffpIUVpUqVQvfu3TFlypS3
auNkKn6c/USvXXXXFM7KTiExkZCV9fX5w6drkLXTpUsI2wzAoLCz
kwoULRKNRUlJSePLkyWyImSTa7ABH5f8rIyMDhcNDU1MSSJUsYHBzk3L
lzE9tdLhc2m41QKMSyZctV9tdLjiQnDA4mA5+Nj1+Ua6e6yJXMwm3mi2kfA7ul1Syy+SM
zE9tdLhc2m41QKMSyZctITU3FbrezcuVKbt++jcfj4dOnT5w5w5c2ZWszU1NTV4vV
5WrFjBoUOH+PbtG8FgkLNNnzybst3nzzZoLBIHv27CE5OZnKykoikQjcJjdu3fjcrmI
RCIMDw+Tm5s759dFROaWZnREx+f7u5u1q1bR2VlZcIsSnSJyJyvX19T
Q2NrJ06VL8fj8j8fj4dO0Z5eTkul2vG8fUFfHw4UNaWlrIy8y8tj586d
E0tlPyssLKS1tZXz589Peno6L6L1++xOfzkZOTaoal/DZ66
kpEREQsSzM6IiIiYllYkKOiIiIiImJZCjoiIiIiImJZCjoiIiIiImJZCjoiIiIiImJZCjoiIiIiImJZCjoiIiIiImJZCjoiIiIiImJZCjoiIiIiImJZfwIvVWjeyDLPywAAAABJRU5ErkJgg
g==",

```
   "text/plain": [
    "<Figure size 640x480 with 1 Axes>"
   ]
  },
```

```
   "metadata": {},
    "output_type": "display_data"
   }
  ],
  "source": [
   "data['normal'].value_counts().plot(kind = 'bar', title = 'Different attarcks distributins',
xlabel= 'atarcks');"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 8,
  "id": "2ae4e738-c461-4599-a654-d09f710c49af",
  "metadata": {
   "tags": []
  },
  "outputs": [
   {
    "data": {
     "text/plain": [
      "normal\n",
      "normal             67342\n",
      "neptune            41214\n",
      "satan               3633\n",
      "ipsweep             3599\n",
```

```
 "portsweep          2931\n",
 "smurf              2646\n",
 "nmap               1493\n",
 "back                956\n",
 "teardrop            892\n",
 "warezclient         890\n",
 "pod                 201\n",
 "guess_passwd         53\n",
 "buffer_overflow      30\n",
 "warezmaster          20\n",
 "land                 18\n",
 "imap                 11\n",
 "rootkit              10\n",
 "loadmodule            9\n",
 "ftp_write             8\n",
 "multihop              7\n",
 "phf                   4\n",
 "perl                  3\n",
 "spy                   2\n",
 "Name: count, dtype: int64"
]
},
"execution_count": 8,
"metadata": {},
"output_type": "execute_result"
```

```
   }
  ],
  "source": [
   "data['normal'].value_counts()"
  ]
 },
 {
  "cell_type": "markdown",
  "id": "943882e1-03eb-452a-827e-8ef7672942e2",
  "metadata": {
   "tags": []
  },
  "source": [
   "The attack types are grouped based on their characteristics and objectives. Here's why each attack belongs to its chosen group:\n",
   "\n",
   "#### DOS (Denial of Service)\n",
   "These attacks aim to disrupt the availability of a service by overwhelming the target with excessive requests or exploiting vulnerabilities\n”.
```

DEVELOPMENT OF AN IMPROVED WEB BROWSER EXTENSION FILTER FOR ONLINE SECURITY

BY

EGENTI, GRACE EBERE
(ACE21150005)

A THESIS SUBMITTED TO AFRICA CENTRE OF EXCELLENCE ON TECHNOLOGY ENHANCED LEARNING (ACETEL), NATIONAL OPEN UNIVERSITY OF NIGERIA (NOUN), ABUJA – NIGERIA

APRIL, 2025

DEVELOPMENT OF AN IMPROVED WEB BROWSER EXTENSION FILTER FOR ONLINE SECURITY

BY

EGENTI, GRACE EBERE
(ACE21150005)

BEING A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF

THE DEGREE OF
DOCTOR OF PHILOSOPHY (PH.D.) IN CYBERSECURITY

AT

THE AFRICA CENTRE OF EXCELLENCE ON TECHNOLOGY ENHANCED LEARNING
NATIONAL OPEN UNIVERSITY OF NIGERIA, ABUJA

APRIL, 2025

DECLARATION

I declare that the work in this thesis entitled '**Development of an Improved Web Browser Extension Filter for Online Security**', has been carried out by me in the Africa Centre of Excellence on Technology Enhanced Learning (ACETEL), National Open University of Nigeria (NOUN) under the supervision of Prof. Okesola Olatunji Julius (main supervisor), Dr. Olalere Morufu (co-supervisor) and Mr. Austin Yusuf (Industrial supervisor). The information derived from the literature has been duly acknowledged in the text and a list of references provided. No part of this work has been presented for another degree or diploma at this or any other institution.

_____	_____
Student's Name	Signature

CERTIFICATION

This thesis titled '**Development of an Improved Web Browser Extension Filter for Online Security,** by **Grace Ebere EGENTI** meets the regulations governing the award of the degree of Doctor of Philosophy (PhD) of the National Open University of Nigeria, and is approved for its contributions to knowledge and literary presentation.

Prof. Julius Olatunji Okesola ……. Date ……..

(Main Supervisor)

Dr. Morufu Olalere ..……..…..……. Date…….

(Co-Supervisor)

Mr. Austin Yusuf ..……..…..……. Date…….

(Industrial Supervisor)

Prof. Grace Jokthan ..……..…..……. Date……..

(Director)

Prof. Sonnie Oniye ..……..…..……. Date…….

Dean, (SPGS)

DEDICATION

This thesis is dedicated to Almighty God for His mercies, love, good health and sound mind. I owe it ALL to HIM alone for bringing me thus far in my academic pursuit. I also wish to dedicate this accomplished milestone in life to the memories of my late brothers and sister, Engr. Obii Ojinkeya, Pastor Gordy Ojinkeya and Mrs. Josephine Nwaneri who passed on before I could complete this study. Their unshakeable faith in God inspired me to persevere until I reached my goal. May their beautiful souls continue to rest in peace, Amen.

ACKNOWLEDGEMENTS

I give special thanks to Almighty God for His ever abiding presence and guidance through this difficult, but incredible journey. I would like to thank my Chief Supervisor, Prof. Julius Olatunji Okesola for his exceptional patience, guidance and support as I worked through numerous iterations of my idea paper, proposal and this final report; Sir, your faith in my ability to complete this dissertation was a tremendous inspiration to me. I also appreciate my co-supervisors, Dr. Morufu Olalere and Mr. Austin Yusuf for being there for me.

Next, I remain eternally grateful to my husband, Engr. Herbert Egenti, for his support and for always asking, “How are you progressing in your research, how far have you gone?” Whenever I hit a roadblock and took a brief hiatus, hearing that question repeatedly motivated me to get back to work. To my lovely children, thank you for the understanding you showed.

My sincere gratitude goes to the director and staff of the Africa Centre of Excellence on Technology Enhanced Learning, especially the Centre Director, Prof. Grace Jokthan, for the scholarship that was granted to me in the course of this research; it supported me in no small measure. My programme coordinator, Dr. Adeyinka Abiodun, I am truly grateful for always checking on me. Dr. Johnson Opateye, you are indeed a father and a great encouragement to me. My brothers from another mother, Mr. Chidiebere Nwankwo, and Mr. Felix Nwagba, I will not forget your genuine concern and your regular questions ‘how far!’ My sincere gratitude goes to Prof. Ishaq Oyebisi Oyefolahan, who doubles as a Research Professor with ACETEL, and the internal examiner for this work. Thank you all.

My gratitude goes to my elder sisters; Mrs. Joyce Amaechi (Adanne, like no other), Mrs. Justina Ugwonno, and Ms Vera Ojinkeya. I couldn’t have asked for better sisters. Thank you for always remembering me in your prayers. I would like to mention some of my course mates, 2020/2021 pioneer set of PhD Cyber Security class that supported me in no small measure, Mr. Ajayi John, Mr. Talabi Doyin, and Mr. Victor Effiong.

Finally, I remain grateful to the Almighty God from whom all blessings flow and with Him all good things are possible, for crowning my efforts to make this exercise a success.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

2FA	Two-factor Authentication
ACETEL	Africa Centre of Excellence of Technology Enhanced Learning
AI	Artificial Intelligence
APWG	Anti-Phishing Working Group
API	Application Programming Interface
AT	Activity Theory
CAPTCHA	Completely Automated Public Turing Test To Tell Computers and Humans Apart
CART	Classification and Regression Trees
COVID-19	Coronavirus Disease
CPU	Central Processing Unit
CRUD	Create, Retrieve, Update and Delete
CSEAN	Cyber Security Experts Association of Nigeria
CSFI	Cyber Security Forum Initiative
CSRF	Cross Site Request Forgery
CSS	Cascading Style Sheets
CSV	Comma Separated Value
DB	Database
DICT	Directorate of Information and Communications Technology
DL	Deep Learning
DNS	Domain Name Server
DOM	Document Object Model
DPPR	Data Protection and Privacy Regulation
DT	Decision Tree
EDA	Exploratory Data Analysis
EGENTI-Filter	This is coined from the author's surname, which means 'to listen' in Igbo language. In this case, listening and filtering the activities of online users
FCC	Federal Communications Commission
FN	False Negative
FP	False Positive
FTP	File Transfer Protocol
GARS	Grooming Attack Recognition System
GFW	China's Great Firewall
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	HyperText Transfer Protocol Security
ICT	Information and Communications Technology
IDE	Integrated Development Environment
IP	Internet Protocol
IS	Information System
ISP	Internet Service Provider
JSON	JavaScript Object Notation

KNN	K-Nearest Neighbor
KPI	Key Performance Indicator
Malware	Malicious Software
MFA	Multi-Factor Authentication
ML	Machine Learning
NCS	Nigeria Computer Society
NLP	Natural Language Processing
NN	Neural Network
NOUN	National Open University of Nigeria
NSFW	Not Suitable For Work
PC	Computer System
PDF	Portable Document Format
PICS	Platform for Internet Content Selection
PIN	Personally Identifiable Number
RBAC	Role Based Access Controls
RBAC	Risk-based access control
RDD	Resilient Distributed Datasets
RF	Random Forest
SNS	Social Networking Sites
SMS	Short Message Service
SOCs	Security Operation Centers
SVM	Support Vector Machine
TAM	Technology Acceptance Model
TF-IDF	Term Frequency-Inverse Document Frequency
TN	True Negative
TP	True Positive
TPU	Tensor Processing Unit
TRA	Theory of Reasoned Action
UI	User Interface
URL	Uniform Resource Locator
VPN	Virtual Private Network
W3C	World Wide Web Consortium
WWW	World Wide Web
XAML	Extensible Application Markup Language
XML	Extensible Markup Language
XSS	Cross-Site Script

ABSTRACT

The surge in digital activities reinforces the need for the protection of online users from malicious, unsafe and unwanted content as well as data breaches. With such a need, web content filtering solutions have evolved as an efficient way of ensuring online security by regulating web access through rules and policies, and other approaches, such as blocking undesirable websites. These techniques help in protecting sensitive data, increasing employee productivity output, and improving user experience through minimization and in some cases elimination of exposure to dangerous online material. Ensuring a safe web environment is still a challenge, as new threats evolve and the amount of new content created each day is quite high. The problem lies within the structure of many existing systems and the approach to users: the focus is primarily placed on adequate protection with little relevance accorded to user experience. Often times, online security relies heavily on filtering which fails to show desired results in times when urgently needed (real-time). Even so, different approaches that include the use of machine learning and database systems have been explored in previous studies to improve web content filtering methods. As much as these approaches claim to be useful, such researchers still maintain that their work does not provide the level of scalability, accuracy, customisation and real-time performance that is needed in dynamic web environments. Some of the identified weaknesses in the previous solutions relate to the capability to adequately handle and process large volumes of web traffic, manually updating the database, the failure to take into account the end user's personalization aspects, as well as the challenges of content over-blocking and under-blocking. Moreover, the majority of the already developed solutions were built with a reactive stance rather than an anticipatory one, therefore, not so helpful in preventing new threats that emerges on a daily basis. This study developed a real-time web content filtering model, known as Egenti-Filter using the developmental research design methodology. It utilizes artificial intelligence to improve the model's flexibility, scalability and user-oriented approach. With Egenti-Filter some of these shortcomings were addressed, thus increasing users' online security and improving users' experiences by intercepting URLs containing malware even before the users come into contact with the URLs. After deployment, a few phishing websites were accessed, and they were blocked by the browser extension, therefore confirming the usefulness of the developed browser extension, which offers a proactive and efficient approach for safeguarding online activities in a transforming cyber-space. With an intuitive Graphical User Interface, users can easily set their filtering preferences and reconfigure controls. With this interface, users can more easily redefine filtering schemas based on their preferences or internet policies. The web content filter ensures that users could timely and effectively navigate and manage their online activities in accordance with the defined rules by providing such a degree of customisation. This versatility increases user experience and simultaneously enhances the strength of the filter in the prevention of accessing malicious URLs. An accuracy rate of 83.83% was achieved with our model.

CHAPTER ONE
INTRODUCTION

1.1 Background to the Study

The Internet is widely used in our day-to-day life since virtually every aspect of human activity has a pinch of Information and Communication Technology (ICT), making the society more digital as lots of businesses, discussion, debates, and dialogue are beginning to take place online, and this era of unmatched connectedness and information exchange was brought about by the development of the Internet in the latter half of the 20th century. However, this has brought about risks and vulnerabilities associated with this revolutionary transformation, especially in the form of malware, phishing schemes, and other cyber-attacks (Nagar & Manoharan, 2024). At first, static methods, such as the signature-based detection systems were the primary tools for combating malicious activities; however, these methods were not very effective at handling new and developing attacks because they relied on pre-existing patterns to identify threats but often failed to address novel and evolving attacks (Abiade, 2024; Alanazi et al., 2023). Furthermore, with the advancement in ICT technologies, the use of social networking sites has increased and become part of our daily life. Also, the proliferation of smartphones has increased online activities to the extent that it is impossible to ignore a notification to check one's social media accounts for updates on friends, family, official, and even personal activities (Cobbe, 2021; Venugeetha et al., 2022).

As the Internet has become more complicated, dynamic techniques like behavioural-based models and heuristic analysis were created to handle increasingly complex threats (Maalem Lahcen et al., 2020). These improvements represented important turning points in cybersecurity, especially as it concerns real-time web content filtering. Meanwhile, the widespread use and acceptability of online activities is highly exposing users to internet security issues (Perera, 2021) thereby raising security concerns to the stakeholders and attracting the interest of many researchers. However, non-technical barriers such as lack of user engagement, inflexibility, and scalability issues have hampered progress in this field. Innovations like the model developed in this study, known as the Egenti-Filter have been made possible by the growing usage of Artificial Intelligence and machine learning, which has created new opportunities for creating

intelligent, flexible, user-centric solutions, to develop sustainable and robust user-focused solutions and products.

The concept of web content filtering is one of the aspects of cybersecurity practices used in securing online users and devices (Federal Communications Commission, 2020; Hyeon, 2023), Web content filtering is a process used to restrict or control access to certain websites or online content. It has transitioned from rudimentary block-lists to AI-driven intelligent systems capable of predicting and preventing threats before they materialize (Abiade, 2024). The integration of artificial intelligence into this domain represents a shift from reactive to proactive security measures. As cyber-attacks continue to evolve, there is a growing need for models that are not only accurate and effective but also flexible and user-oriented. According to (Ojo, 2024), in addition to scanning the domain name, a modern and efficient web content filtering solution may also analyse and dissect web traffic, allowing it to precisely identify parts of a page that should not be allowed to get into the internal network.

According to Turner, (2022), a system is evaluated on how efficiently it uses resources while doing a specific activity or meets a technical criterion, and how well actual users are restricted from accessing prohibited content on the web. However, past literature studies on the effectiveness of web content filters were known to block questionable content on the internet and are even focused on small, convenient samples (Diaz-Garcia et al., 2022). The review of filtering software was also based on the personal evaluations of web filtering software that were used in the last few years, rather than on scientific investigations (Zeebaree et al., 2020). Most of the challenges that surround web content filtering technology therefore come from the approaches that are employed to filter/block access to web content (Clarke, 2019).

No matter the techniques adopted, it is generally believed that filters are not an absolutely perfect solution to the issues of web content security (Turner, 2022). Filtering technology has some challenges, some either over-blocks and/or under-blocks web content, thereby denying access to legitimate information and permits access to inappropriate contents (Diaz-Garcia et al., 2022). As opined by Turner, (2021), considering the Packet blocking approach at the Internet Service Providers (ISP) filtering level, the implementation of filters can be done in different ways, such as firewalls that are to be deployed on the connections, and having the comprehensive list of websites to be blocked, this is an advantage. However, a major challenge with this filtering

approach is the collateral damage that comes with it - all the known web contents on a specific IP address could be blocked, therefore, making it inaccessible to legit users (Mcmanamen, 2018).

The author believes that deployment and use of content filters can be made more efficient with a greater understanding of the applications and constraints of filtering approaches. Hence this study primarily aims at developing an effective real time web content filtering model for an improved users' online experience.

1.2 Statement of the Problem

As the Internet becomes an integral part of our daily lives, there is growing concern over the security of online activities, such as online shopping, social media interaction, email communication, etc. With the vast amount of content available online, it is difficult to ensure that users are not exposed to harmful or inappropriate material, these exposures has caused an increase in cases of cyberbullying, child pornography, human trafficking, online gaming, identity theft, addiction, and many more attacks (Lan et al., 2022) and the overall effects have been devastating for online users.

Phishing, malware, and credential theft are all growing at a rapid pace, e.g., According to the Anti-Phishing Working Group (APWG) 2023, there were 4.7 million phishing attacks in 2022 - the most ever recorded in a single year and 65% more attempts in credential theft. The global cost of cybercrime was therefore anticipated to surge dramatically, escalating from $9.22 trillion in 2024 to a staggering $13.82 trillion by 2028. This alarming trend underscores the urgent need for heightened vigilance and innovation in cybersecurity strategies (Statistica, 2024). Different reports (Khoo et al., 2020; Mathews & Chimalakonda, 2021; Diaz-Garcia et al., 2022) are beginning to link these attacks to the unguided use of Internet Technology due to knowledge gaps on the social networking users. However, Clarke, 2019; Okay, 2023 argued that the lack of right approaches to monitor the activities of online users and expressions is a greater challenge.

At first, static methods (signature-based detection systems) were the primary tools for combating malicious activities. However, these methods were not very effective at handling new and developing attacks because they relied on pre-existing patterns to identify threats but often failed to address novel and evolving attacks. Thereafter, other approaches (user authentication, network security, and content filtering) have been implemented to secure the online activities of users.

Some approaches such as keyword filtering, IP blocking, Firewalls, Proxy Servers, Users credential authentication (by username and password or PIN), biometric authentication (face recognition, finger print, or voice recognition, among others) have been implemented, and have achieved their successes. However, these approaches have been considered ineffective for high rates of over-blocking and under-blocking, and some of them can easily be circumvented and have no frequent updates, therefore, new attacks cannot be detected (Alanazi et al., 2023). More so, some proxy servers can pose some tracking and security issues, and the single authentication which is characterized with weak passwords that may be easily exploited. Traditional methods applied in the detection of malicious websites, such as blacklists, update less frequently, and as such cannot effectively detect new attackers. Several technologies and methods have been implemented to safeguard digital interactions, but none have turned out to be completely effective for reasons not yet clearly identified (Butun et al., 2020; Nagar & Manoharan, 2024)

Unfortunately, very few studies exist to investigate the accuracy of the available web content filtering solutions (Van Hasselt & Bourke, 2018). These limitations have therefore inclined many researchers to propose more effective security mechanisms such as multi-factor authentication (MFA), intrusion detection and prevention systems, content filtering, etc., to curb cyber-threats (Petru-critian, 2023; Zwilling et al., 2022). There is therefore a need, for Egenti-Filter, a model that balances high detection rates with flexibility, user customisation, scalability, and ease of use for an improved and secure users' online experience. This is what our newly developed Web Browser Extension achieves in this study.

1.3 Research Questions

The research problem stated in Section 1.2 led to the following primary research question:

How can web filters be used to effectively secure online activities of social networkers?

To answer the primary research question, the following secondary research questions are identified and need to be addressed:

RQ1: Which current security control measures are effective for securing online activities?

RQ2: Are web filters effectively securing online web activities?

RQ3: What are the major limitations of web content filters?

RQ4: Why are web content filters considered the best security approach for online users?

1.4 Aim and Objectives of the Study

The aim of this study is to develop an improved real-time web content filtering model leveraging the web browser extension approach for an improved and secure users' online experience.

The specific objectives of this research are to:

1. Identify and understand the current security control measures on social network sites
2. Design a real-time web browser extension filtering model (Egenti-Filter)
3. Instate Egenti-Filter on a secure web browser for effective implementation
4. Evaluate the model performance of Egenti-Filter using appropriate metrics

1.5 Scope of the Study

This study is to develop a model for securing online activities using a web content filtering approach. In designing the model, a web content filter was developed and employed to curate malicious websites and IP addresses. The extracted malicious websites and IP addresses were then used to develop a web content filtering based model that can be used to secure online activities. The developed model was translated to a web browser extension that could be incorporated into web browsers for the purpose of securing online users. Only online activities were explored in this study.

This browser extension guards against phishing and credential theft, prevents malware infections and data breaches, mitigates new and zero-day threats, and improves the user's overall security posture by proactively analysing and blocking suspicious URLs before the user accesses them. By using a preventive rather than a reactive strategy, possible online risks are eliminated before user safety is jeopardized.

1.6 Significance of the Study

This study shall benefit parents, caregivers, schools, organisations, and public libraries, desiring to deploy web content filters by shielding online users from inappropriate content. It will also prevent unauthorized use of network resources such as playing online games, download or watching of adult videos on a work/school network. For businesses looking to increase productivity and reduce liability, content filtering is crucial to safeguard students and university

faculties from harmful websites, educational institutions and libraries to have rights and legal obligation to use filtering measures.

This study could be used by businesses to modify how people visit websites through their network, making it a crucial tool for safeguarding users' and customers' data from online dangers including phishing, malware, and other threats. The control of bandwidth usage maintains employee productivity, and limits the organisation’s exposure and liability by restricting access to inappropriate content will be some of the benefits that could be derived from this study. It will also promote productivity and employee efficiency by ensuring that workers are checked through the filtering of websites during working hours to ensure maximum output during working hours within the organisation.

It is intended to provide data for scholarly reference for web content filtering literature by showing how, and what kinds of filters exists, limitations that could arise in using web filtering which will add relevant information to the body of literature, thereby completing and enriching research on the topic. By installing the developed web browser extension, different categories of users would significantly enhance their online security and protection against cyber threats and other cyber related crimes. Listed are some of the benefits to different categories of users:

Learners

i. Protects learners from accessing inappropriate or harmful content online
ii. Helps educational institutions comply with internet-usage regulations
iii. Prevents distractions, enhancing focus on educational tasks
iv. Safeguards against cyber threats and malware, ensuring a secure online learning experience

Employers

i. Blocking malicious websites and preventing data leakage, safeguarding sensitive information
ii. Reducing legal liabilities
iii. Monitoring employees' online activities and preventing harmful content sharing
iv. Restricting access to non-work-related websites, reducing distractions

Parents

i. Shields children from accessing inappropriate or harmful online content thereby promoting a safe online environment
ii. Helps parents enforce rules and guidelines for internet usage, ensuring children's online safety
iii. Prevents exposure to cyber threats and malware, protecting children's devices and personal information
iv. Allows parents to monitor and control their children's online activities, promoting responsible internet usage.

E-commerce Websites

i. Helps for the identification and blocking of fraudulent activities
ii. Help to enhance customer trust and increase loyalty
iii. It ensure that users are connected to secure payment gateways
iv. It promotes secure shopping experience to customers by avoiding distracting popups

For the Elderly

i. Many elite elders connect with friends and family using social media platforms, hence the browser extension ensures that personal information is protected and their interaction is safe
ii. They feel more comfortable engaging with the online community and accessing services that enhances their quality of life.

1.7 Definition of Terms

Blacklisting: Blocking access to specific websites or content while allowing access to everything else.

Censorship: Censorship refers to the deliberate suppression or control of information, ideas, or media content by governments, organisations, or authorities.

Circumventor Sites: These are parallel websites that allow online users to find a way around filtering software and access sites that have been blocked.

Content Filtering: The process of examining or blocking online content based on pre-defined criteria, such as keywords, categories, or URLs.

Cyberbullying: This term refers to any behaviour exhibited through electronic or digital media by individuals or groups that repeatedly communicate hostile or aggressive messages intended to inflict harm or discomfort on others.

Cybersecurity: This refers to any known or unknown technique, approach or software that is employed to protect computers and computer users and to prevent online crime.

DNS Filtering: Domain Name System (DNS) filtering involves controlling access to websites based on their domain names.

Filter/Filtering: This is an approach that allows users to block certain types of content from being displayed on their device, for example, curse language, nudity, sexual content, and violence.

Firewall: This security system is usually made up of hardware and software tools that are used to block hackers, viruses, and other malicious threats to users' computer.

Internet Service Providers (ISP): A general term for any business that may link you directly to the Internet is an Internet Service Provider.

Keyword Filtering: Filtering content based on words or phrases is known as keyword filtering.

Malware: This is short for malicious software or code, refers to any destructive code such as Trojan horses, worms, spyware, adware, among others, that is intended to harm a computer or gather data.

Monitoring Software: These are software tools that enable parents to keep an eye or track the websites or emails that their children browse or read or visit.

Password: A secret phrase or number that is required to enter a website, access a parental control feature, or alter software**.**

Phishing: This is a type of online fraud, in which an attacker tries to trick the victim into revealing sensitive details, for example, a username, password, or credit card details, and this is done by masquerading as a trustworthy entity in electronic communication.

Proxy Server: A proxy server acts as an intermediary between a user and the internet. It can be used to bypass internet filters or access restricted content by routing the user's requests through the proxy server, which can be located in a different location or have different access privileges.

Real-time: Live time; the actual time during which something takes place.

Real-time Filtering: Real-time filtering involves the immediate analysis and filtering of internet content as it is being accessed or transmitted.

Search engine: An online tool that facilitates online information research.

Social media: Online groups usually referred to as social networks, where members exchange personal information, media files, pictures, among others.

Uniform Resource Locator (URL): The address of a website on the internet is known as the URL. For instance, the ACETEL website is https://actel.nou.edu.ng, while the National Open University URL is https://nou.edu.ng, and personal URL like https://egentigracestores.com.

URL Filtering: Filtering web material based on specific URLs or website addresses is known as URL filtering.

Virtual Private Network (VPN): A virtual private network, or VPN, is a technology that establishes a private and secure connection across a public network like the internet.

Web or the World Wide Web: a hypertext-based Internet navigation system that enables quick access to a range of connected sites by typing commands or clicking on hot links.

1.8 All Chapter Overview

This section presents an overview, of all the chapters in this research endeavour.

Chapter one handles Introduction which makes up of the statement of the problem, the aim and objectives of the study, the scope of study, the significance of study, and concluded with definition of terms. Chapter two focuses on the review of related works, which covers the overview and theoretical foundation guiding this study. The research methodology employed for this study and the phases of the study towards achieving the research objectives are presented in chapter three.

Chapter four handles the results, while chapter five is focused on the discussions of the findings of the study are presented and validated to authenticate how the Research objectives are achieved. Research reflection and conclusion are presented in chapter six; the entire study is summarised, reflected on and conclusions are drawn with recommendations for future study.

CHAPTER TWO
LITERATURE REVIEW

2.1 Preamble

There have been a lot of challenges for online social network users, such as data security and privacy issues, and the exposure of unsolicited objectionable content (Anderson, 2013; Bocar & Jocson, 2022; Persia & Auria, 2017). This has necessitated the monitoring of the activities of social networking users. However, there has been a plethora of study (Bandari, 2023; Koohang, 2021; Perera, 2021; Wong & Liang, 2021; Turner, 2022) on the monitoring of the activities of online content using different approaches, such as whitelists and blacklists, URL-Based filtering, AI content filtering, DNS-Based filtering, Keyword-Filtering, IP and Protocol-Based Blocking, Dynamic Filtering and Platform-Based Blocking. However, the problem has been traced to the effectiveness of the right approaches of filtering web content (Lausanne, 2017).

After the Covid-19 pandemic era, the research conducted by De et al. 2020; Omeluzor et al. 2023), demonstrated the importance of technology for post-COVID-19 recovery and growth. Online users such as students, small and large business administrators, teachers, private and public employees, and every online user can learn about digital citizenship to better understand how to utilize technology. More so, because there are so many news and media sources available, there is a rise in fraud, deceit, and misinformation; for this reason, online users need to choose trustworthy sources, and be sure of the content they are accessing (Kozyreva et al., 2020; Sheth, 2020).

Figure 2.1 shows Internet use over time, which revealed that at the start of the 4th quarter of 2023, around the world, 5.30 billion people or 65.7% of the total population was internet users. More so, Internet users are also increasing, with the most recent data showing that the world's connected population increased by 189 million users in the year ending October 2023 (Kemp, 2023). Although this 3.7% annual growth rate is a little lower than the growth rates observed in the middle of the previous ten years, it is still among the higher rates observed in recent months.

Figure 2.1 x-rays the essential digital headlines indicating that the global population without internet access has dropped to 2.77 billion, with most of them residing in Africa and Southern and Eastern Asia (Simon Kemp, 2023). This implies that much effort needs to be done before

everyone in the globe could have access to the Internet and that people's internet connection quality is a crucial factor. However, current trends indicate that by the middle of 2024, the vast majority of internet users worldwide, 95% use their mobile phones to access the Internet at least occasionally, and mobile phones currently account for around 57% of the online time, and hence, two-thirds of the world's population will be online (Kemp, 2023).

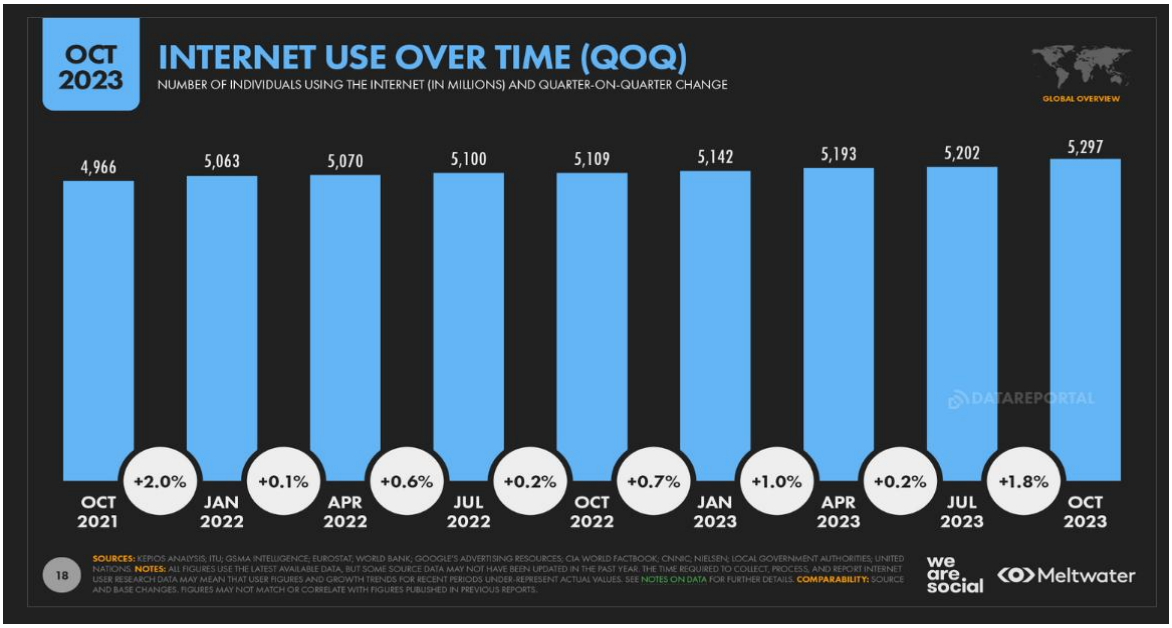


Figure 2.1: Internet use over time (Report, 2023)

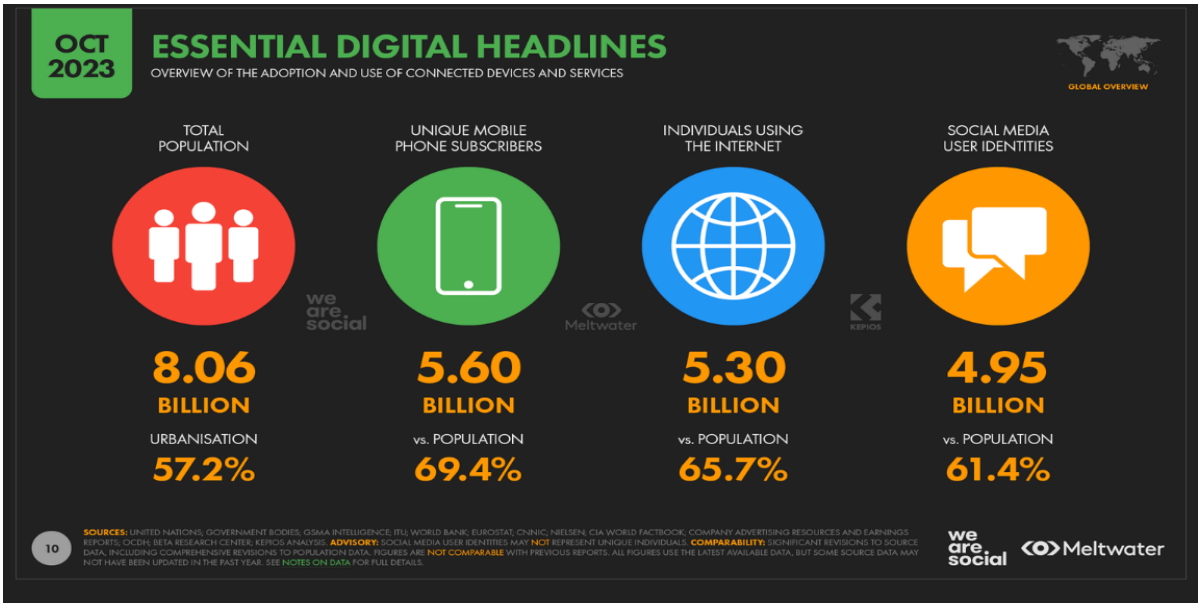


Figure 2.2: Essential digital headline (Report, 2023)

2.2 Initial Theoretical Foundation

Any research project's theoretical framework is a thoughtfully crafted and cohesive set of concepts or principles drawn from theories that are solely employed to support a study. It seeks to make clear the existing hypotheses that are relevant to the on-going research and how they relate to one another. It also draws attention to the theory or theories that form the basis of the research (Salawu et al., 2023). The theoretical framework is a logical argument that connects the individual rationales and explains why researchers' predictions are the best the field can make right now, and the coherent argument built from the rationales developed as part of each hypothesis that was formulated is referred to as the theoretical framework (Cai et al., 2023). For the fact that each rationale explains why any particular prediction was made, it also contains useful cues for which the methods would provide the most fair and comprehensive test of that prediction. Indeed, the theoretical framework provides a logic against which one can test every aspect of the methods that is planned to be used (Adom & Hussein, 2018).

A theoretical framework should typically include the following three basic features (Hohensee, 2023; Okesola, 2014),which shall be related to this study in the next sections:

i. A description of the variables deemed relevant to the research.

ii. A conceptual model that explains how the variables relate to one another.

iii. A concise justification for the anticipated existence of these connections.

2.2.1 Relevant Variables to this Study

The Web browser extension of any known browser will not be improved upon if the effectiveness of the other fundamental components is not sufficiently measured. The Web Browser Extension is the subject under study and it is regarded as the global theme for this study. This global theme is a dependent variable as the effectiveness of the web browser extension is dependent on so many factors thereby considered independent variables in this research endeavour. This include the filter's graphical user interface (GUI) design, the Database Management module, the web crawler application, the URL classifier, the machine learning module and the entire web browser extension framework. Although, web browser extension (the global theme) is dependent on the independent variables in the research; the latter is independent on any factor.

2.2.2 The Conceptual Framework Relating the Variable

The conceptual model for this study that relates both the dependent and independent variables which influences the effectiveness of the security of online activities using the browser extension approach (Figure 2.3).

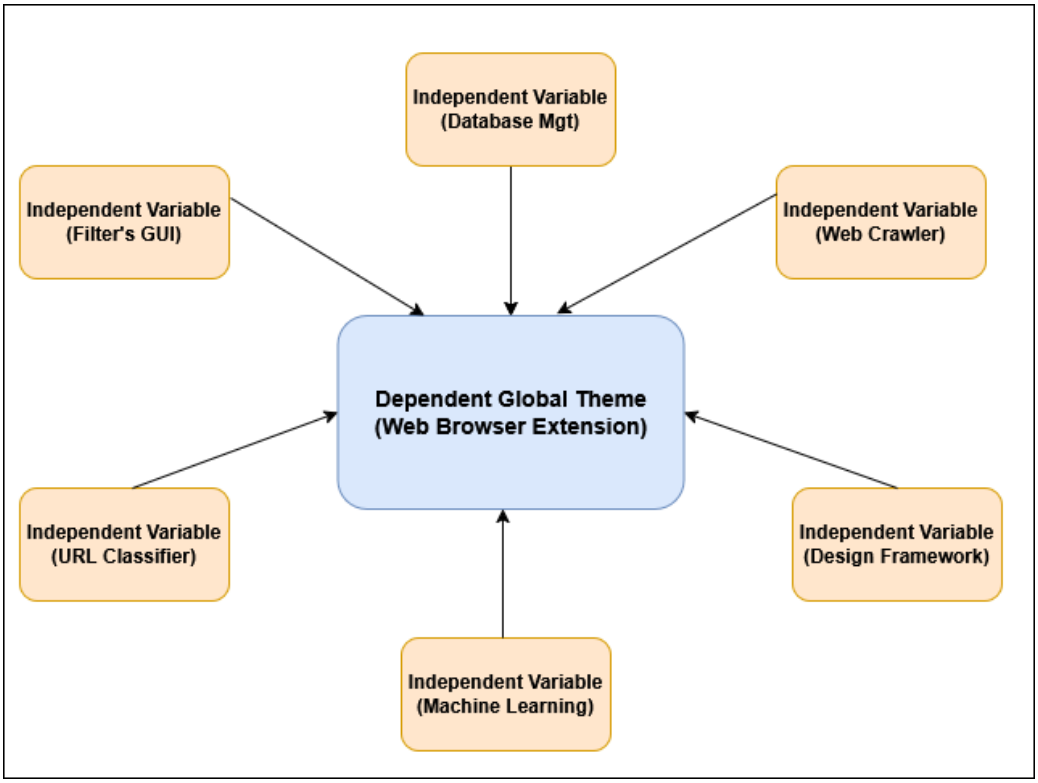


Figure 2.3: Initial framework for Web Browser Extension

2.2.3 Claims for the Expected Relationships to Exist

Some past related studies (Diaz-Garcia et al., 2022; Voros et al., 2023; Yu et al., 2020) discussed in subsequent sections identified the attributes highlighted as the fundamental components and potential factors influencing the effectiveness of a web browser extension, and are therefore considered as independent variables. The web browser extension is the dependent variable because it depends on the effectiveness of the web crawler, the database management system, the URL classification and the Graphical user interface (GUI) for updating of the database. Hence, the success or failure of the global theme is highly dependent on the independent variables. The web crawler is expected to effectively crawl websites and/or documents to extract links; while the URL classifier should be able to classify the URLs into malicious or benign links, and the

filter's GUI is to update the database. This ensures that the global theme performs effectively, and this is dependent on the other components. The URL classifies the contents of the CrawledLinks into malicious or benign, and the classified malicious links are temporary stored in the Classified Malicious file, where it is used by the GUI to update the database. It is important to note that the vectorizer (to convert the textual data into numeric data that the classifier can understand) and the model are used by the URL classifier to perform the classification operation. The entire application is packaged in a way that users do not have to run install on their device for the application to be installed, rather, simply copying and pasting to a preferred location.

2.3 The Referent Theories

This study on Browser Extension for online security is an Information security concepts but with a feature on human behaviour as several researchers (Khoo et al., 2022; Kuraku et al., 2022; Mathews & Chimalakonda, 2021) are beginning to link security attacks to the unguided use of Internet Technology due to knowledge gaps among social networking users. The referent theories for this study are the three theories developed by Markus (Myers & Avison 2002): people-determined, interaction-determined, and system-determined. Other similar model Activity Theory (AT), Theory of Reasoned Action (TRA), and Technology Acceptance Model (TAM) was also adopted.

2.3.1 System-determined, interaction-determined and people-determined theories

System-determined, interaction-determined and people-determined are the adapted referent theories for this research, which represents Markus's three theories (Myers & Avison, 2002). Meanwhile, 'the rudimentary assumptions fundamental to the theories can be studied and likened with facts in the real-world'. Markus' competing theories of system and people's resistance or otherwise to study the impacts of online security threats are the fundamental assumptions that are essential to online security and technology usage, the ways of mitigating such threats and for users to have a safer online experience.

i. System-determined Theory

According to Myers and Avison (2002), the system-determined hypothesis attributes resistance or non-utilization to inherent characteristics of the system that has been put into place. This theory's explanations include the fact that people dislike technologically complex systems,

poorly designed ergonomic systems, and unfriendly systems. According to the system-determined hypothesis, a system's design elements, such as its security settings, can decide whether a given system is accepted or rejected in any specific organisation. The system-determined theory's fundamental premise is that system users are inherently non-utilizers.

ii. Interaction Theory

According to the interaction hypothesis, Myers and Avison (2002) opined that a person's qualities and a system's characteristics interact to cause resistance or underutilization of systems. Resistance resulting from the interplay between the technical aspects of the system's design and the social context in which it is employed is one explanation offered by the interaction theory. The fundamental tenet of interaction theory is that system users, system designers, and privacy settings all contribute to non-utilization or insecure behaviour. Different results for system implementations in different circumstances and different responses from the same user group in different situations can both be explained by the interaction theory.

iii. People-determined Theory

The people-determined theory takes into account innate human attributes like human nature, cognitive style, and personality traits. According to the people-determined view, people's resistance to information systems is a result of internal variables that affect them individually or as a group. Resistance is also seen as a feature of system users. When any one system is refused, the theory performs well at forecasting the rejection of all systems (Myers & Avison 2002).

2.3.2 Similar Adopted Theoretical Frameworks

The concept of certain similar model was also adopted, namely: Activity Theory (AT), Theory of Reasoned Action (TRA), and Technology Acceptance Model (TAM).

1. Activity Theory (AT)

AT is a theoretical framework used to analyse and comprehend human interaction through the usage of tools and objects (Hashim & Jones, 2007). In a study conducted by Adamides (2023), the researcher presented the application of activity theory to the understanding of the socio-technical system shift from recorded music to streaming, and to support the regional management of the shift from the production of olive oil to a circular economy by implementing cutting-edge waste-processing technology. In another study conducted by Gogus (2023), and in

order to meet the course's learning objectives and the prerequisites for postsecondary education, the study addresses a framework that was created to adapt Activity Theory for the design, development, implementation, and evaluation of online courses. This framework aims to improve the cognitive, teaching, and social presences within complex cognitive tasks. Activity theory (Figure 2.4) has seven components helping it function well. The components include subject, object, tool, rule, community, division of labour, and output (Andriani et al., 2022).

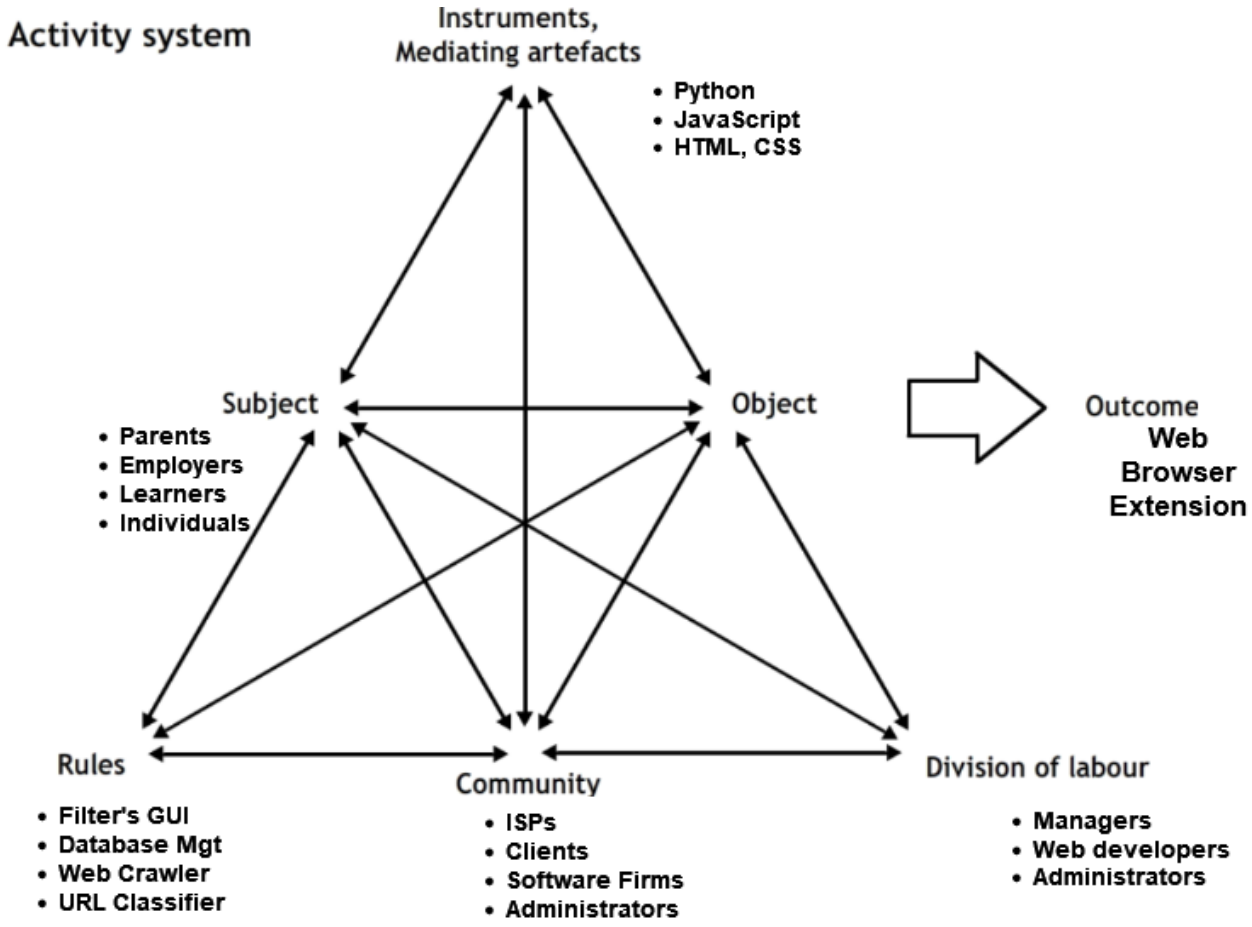


Figure 2.4: Activity theory triangle for this study

i. The Subject

A subject in the Activity Theory is seen as the individuals that are part of the activities and who performs the actions to achieve stated output/result (objectives). However, for the subjects to attain the expected outcomes there is the need for the members of the community to effectively carry out their roles. Engeström (2015) extended Leontiev's (1978) concepts to a triad-connecting subject, object, tools and widened the system to contain community, the people who

share the same intention, rules ruling over the community, and division of labour that facilitate collective actions that individuals are involved in.

ii. The Object

The intention why the activity takes place is majorly seen as the object (ive) is the problem. It is also viewed to be a problem-space which is to be converted (transformed) or molded by the activity. In essence, the object is the activity being acted on to reach a specified objective; in this case, it is the design and development of a web browser extension for securing users' online activities.

iii. Outcome

The outcome is the overall goal that the activity subjects are expected to accomplish; these activities are targeted to achieve the goal. All of the activity system plays a different role to actualize the primary objective (Postholm, 2015). Therefore, the outcome of the activity is the expected outcome of carrying out the activity.

iv. Tools

Tools/instruments are the means such as the technological artifacts or other softer elements (language and signs) by which the activity is being carried out (or executed). In this case, the tools are the instruments or techniques that were used to design the Web Browser Extension, such tools are the Python programming language, JavaScript, HTML and CSS.

v. Rules

The rule represents the formal and informal rules, that is, institutions and practices that administer the transformation of the primary objective, for example, the experience of the IS manager, the web developer, the network administrator, and other processes for achieving the result. The rules are instituted to target and manage the activities. Without these rules, there may be clashes of interests that may hinder/delay the achievement of the intended outcomes. Furthermore, it is intended to prevent impediments and maintain order (Postholm, 2015).

vi. Community

Community represents the different stakeholders involved in the activity. In the community, the activity subjects work harmoniously together to realize the primary goals. The community tends to allow the cooperation with diverse sets of labour divisions which are dependent on the need to achieve the objectives.

vii. Division of Labour

The section, division of labour indicates who does what, and how roles and power orders are organised. In essence, the division of labour outlines how tasks and power are circulated among community members, all geared towards achieving the primary objective of designing and developing a web browser extension. In conclusion, Adamides, (2023), stressed that the elements of 'community' and 'division of labour' of the high-level activities help identify lower-level activities that contribute significantly to change. These particular tasks, roles, and power dissemination allocated in the upper-level activity's division of labour define the objectives of lower-level activities, which are seen as topics of the community of higher-level activities.

In summary, the subjects represent the Web Browser Extension Users who need to secure their online activities, this category of users include parents, organisations and individuals. The Object is the Web Browser Extension Filter (i.e. the desired output). The Rules are the different components of the Output, such as the Filter's Graphical User's Interface (GUI), Web Crawler Application, URL Classification, and the Database Management Module; while the Tools are the instruments or techniques that were used to design the Web Browser Extension, such tools are the Python programming language, JavaScript, HTML, CSS, etc. The Community refers to the Web Browser Extension stakeholders (ISPs, IS managers, etc.) who are interested in online security, while the Division of Labour refers to the Organisation or IS manager who owns the Web Browser Extension and are duly responsible for its effective internal workings.

2. The Theory of Reasoned Action (TRA)

Theory of Reasoned Action (TRA) stems from the idea of social psychology, known as the famous theory of human behaviour (Morrow & Morrow, 2022; Okesola, 2014;). Of the three classic models of persuasion in psychology and health behaviour, it was first to be developed. This theory has presented a framework for the discovery of some basic processes that defines human behaviour across various social settings, as it provides the connection between behaviour and attitudes; a classic topic of study within social psychology, and a pivotal topic of discussion (Hagger, 2019). Hagger opined that the fundamental idea of this theory is deliberate, which is the driving factor and seen as the most proximal factor of behaviour. According to Godin (2021), in the theory of reasoned action (TRA), attitudes and subjective norms influence behavioural

intentions, which in turn influence behaviour. This theory is useful in forecasting behavioural variability under various settings, demographics, and actions (Hagger, 2019).

The Theory of Reasoned Action postulates that the behaviour of an individual is motivated by the behavioural intention which is a function of an individual's attitude towards the behaviour and subjective norms which surrounds the performance of such a behaviour (Junaid-ur-Rehman, 2022; Momani, 2020; Salloum et al., 2021; Shen et al., 2020). In essence, it is of the opinion that an individual's behaviour and the resolve to behave are directly proportional to the individual's attitude toward the behaviour and their perceptions about the behaviour.

Deliberate actions show the degree to which an individual or group of persons is likely planning to achieve something, and so, efforts are invested to pursue any given behaviour (Junaid-ur-Rehman, 2022). Again, behavioural intention is used to describe the driving force that impact a certain conduct; the stronger the conviction to carry out the activity, the more likely it is to be carried out. Figure 2.4 shows TRA, which proposes that behavioural intention is the most proximal predictor of behaviour. Behavioural intention is directly determined by the attitude towards involving in the action and other subjective norms related to the conduct. The Theory of Reasoned Action sets itself apart from its forebears by distinguishing between several attitudes and integrating subjective norms as a focal predictor of behaviour (Junaid-ur-Rehman, 2022).

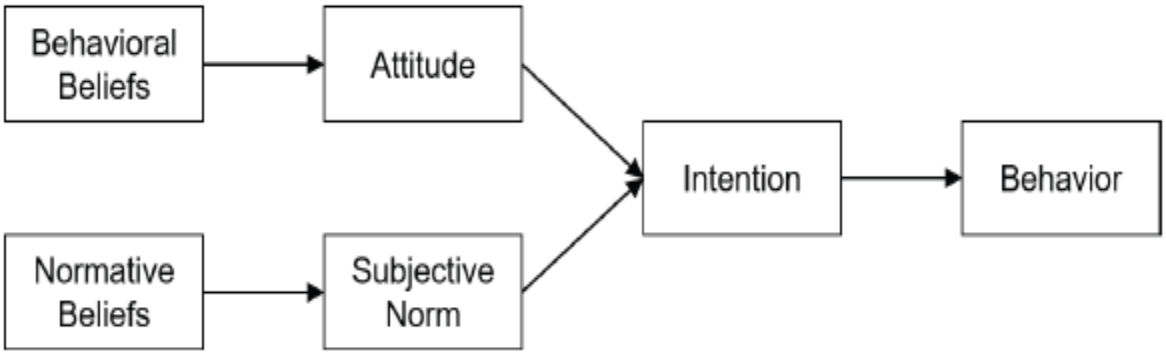


Figure 2.5: The theory of Reasoned Action (Junaid-ur-Rehman, 2022)

3. Technology Acceptance Model (TAM)

Technology Acceptance Model (TAM) is one of the most well-known models for predicting technology adoption and usage behaviour as has been thoroughly proven (Al-Adwan et al., 2023). The model used for previous research was based on TAM. Davis (1989) was the first to create TAM in accordance with the Theory of Reasoned Action (TRA) in the research of

psychology. Technology Acceptance Model proposes that perceived ease of use and perceived usefulness of technology are predictors of user attitude towards using the technology, subsequent behavioural intentions and actual usage (Alshurafat et al., 2021). It was also considered that perceived ease of use influences the perceived usefulness of technology (Figure 2.6).

Specifically, researchers (Salloum et al., 2021; Shen et al., 2020) have been using the TAM more frequently to predict students' adoption of learning technology. However, TAM only provides general insights into users' propensity to embrace technology; therefore, for context-based knowledge of the usage of a specific technology, other elements that may influence a user's adoption of technology are required (Zhang et al., 2022).

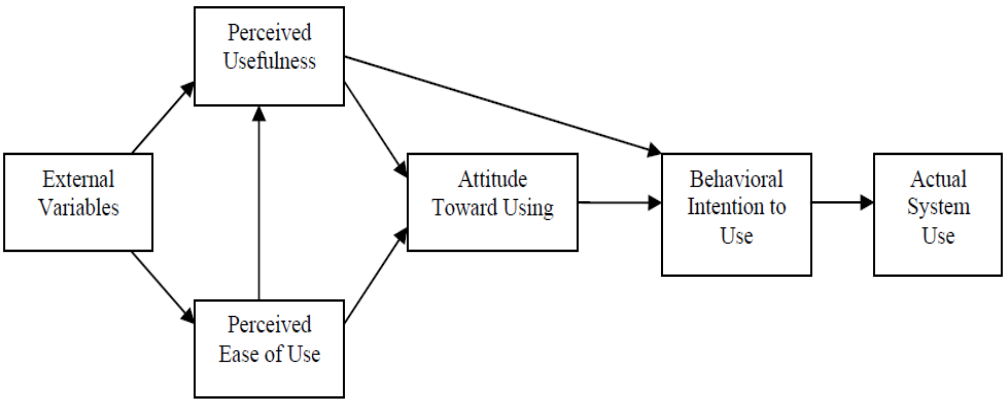


Figure 2.6: Technology Accetance Model (Al-Adwan et al., 2023)

2.3.3 Relationship between the Theoretical Frameworks and the Web Browser Extension

For the Activity theory, the developed web browser extension is the tool of mediation between the online users and the goal of online security. Users' decision to deploy the developed web browser extension is significantly influenced by the attitude they possess towards online security (Theory of Reasoned Action). The adoption of the web browser extension depends on whether the users perceive the extension to be easy to use (not requesting high technical skills), and the perception of usefulness in enhancing online security and protection (Technology Acceptance Model). By adopting these theories, an all-inclusive frameworks is achieved that does not only explain why user may be willing to adopt and use the developed web browser extension, but also offers insights on how to improve the design and effectiveness of the extension; for example, in developing the extension, we could focus on enhancing the perceived ease of use (Technology acceptance model), while highlighting its usefulness. Furthermore, designing the extension to

seamlessly fit into user's online activities (Activity theory) will significantly increase the adoption.

2.4 Conceptual Framework for the Study

The swift development of web technologies has resulted in a growing dependence on browser extensions to improve functionality and user experience. However, there are serious security dangers associated with the spread of malicious extensions, such as phishing attempts, data breaches, and illegal spying. The creation of a better web browser extension filtering mechanism for online security is the main goal of this work. The study's conceptual framework (Figure 2.7), which integrates current cybersecurity theories, filtering techniques, and artificial intelligence-driven security models, forms the basis of the investigation and suggests an improved filtering approach. This conceptual model graphic shows how the various essential elements work.

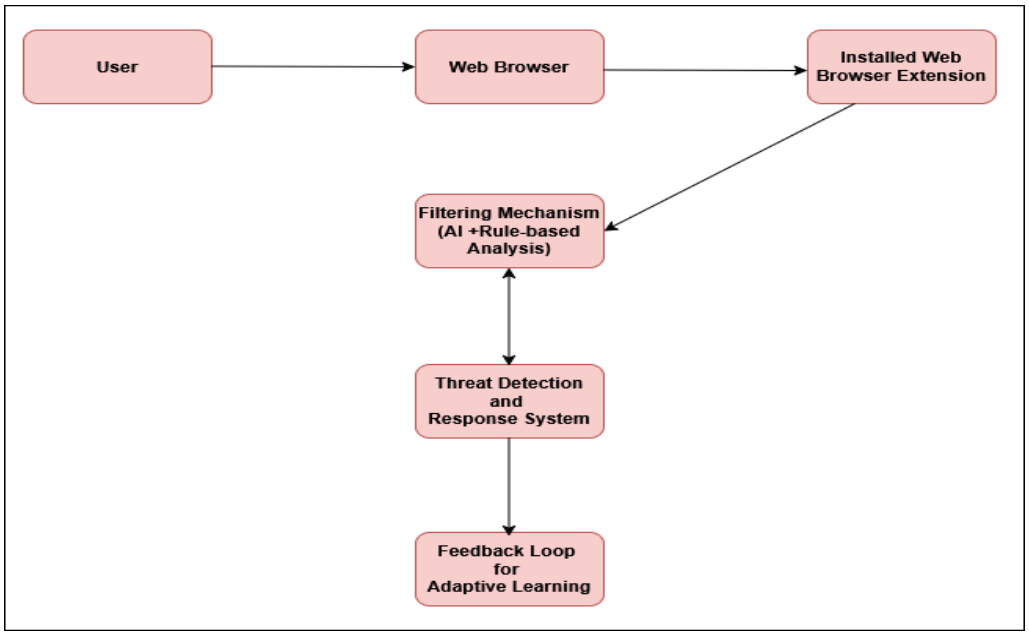


Figure 2.7: Conceptual Framework: Improved Browser Extension Filtering

i. User: The user is a representation of the person using the web browser. Although users may install dangerous browser extensions unintentionally, many do so to improve their browsing experience.

ii. Browser: The main interaction between the user and installed extensions is the web browser.

iii. Installed browser Extensions: Browser extensions that improve functionality, such productivity tools, password managers, or ad blockers.

iv. AI + Rule-Based Analysis Filtering Mechanism: The primary security layer, which uses rule-based and AI-driven filtering to identify and stop dangerous content. makes use of both dynamic and static analysis to monitor runtime activity and validate URLs

v. System for Threat Detection and Reaction: Uses URL classifier to identify malicious URLs When a potentially dangerous URL is identified, the proper security measures are taken (e.g., notifying the user, and access denied from further visiting the URL).

vi. Feedback Cycle in Adaptive Learning: This ensures that the filtering system is continuously improved by examining emerging risks. It adapts machine learning models and filtering rules in response to user feedback and changing threat tactics, which improves detection accuracy, and helps lower false positives and false negatives.

2.4. Web Content Filtering

Web content filtering is one of the aspects of cybersecurity practices used in securing online users and devices (Federal Communications Commission, 2020; Hyeon, 2023), as shown in Figure 2.8 cybersecurity technologies. Web content filtering is a process used to restrict or control access to certain websites or online content. It is typically done by organisations or individuals to restrict access to inappropriate or harmful material, such as adult content, violence, or hate speech (Gómez Hidalgo et al., 2019; Narwal, 2021). Web filtering can be implemented through various methods, such as using software applications, configuring network settings, or utilizing third-party services (Chyrun et al., 2019; Johanna, 2019; Maserumule, 2020; Zhang et al., 2019). It is often used in educational institutions, workplaces, and homes to ensure a safe and productive online environment (Hidalgo et al., 2019; Kova, 2020).

More so, web filtering refers to a type of web content filtering technique employed by organisations and individuals as fragment of an Internet Firewall to determine whether an incoming data is harmful to the network or if outgoing data holds any intellectual property (Clarke, 2019; Khando et al., 2021). Web filtering works by analyzing the content of websites and web pages, as well as the URLs and IP addresses associated with them (Aktay, 2018; Jain et al., 2021; Mienye & Sun, 2022). It helps prevent access to malicious websites, restricts access to

some aspects of categories of web content, and ensures compliance with legal and regulatory requirements (Smith, 2019; Alexei & Alexei, 2021; Etuh et al., 2021; Jain et al., 2021; Herath et al., 2022). Additionally, web filtering can be used to increase productivity by blocking access to distracting or non-work-related websites (Baishya & Kakoty, 2019; Herath et al., 2022).

Figure 2.8: Cybersecurity Technologies (https://www.shutterstock.com/)

Web content filters can be deployed on both individual devices and network infrastructures (Lee & Kim, 2020; Duncan & Chen, 2023). Individual device-based web filtering involves installing software or applications that block or filter out specific websites or content categories. Network-based web filtering, on the other hand, is implemented at the network level and applies to all devices connected to that network; this method allows for centralized control and management of internet access, making it easier to enforce restrictions and monitor online activities across multiple devices (Ali et al., 2021; Chrysomalidis et al., 2021; Jain et al., 2021, Raman 2021).

There are different web filtering techniques that can be employed, such as keyword filtering, which scans for specific words or phrases that may indicate inappropriate or prohibited content; another technique is blacklisting (Ali et al., 2019; Thangaraj, 2019), where a list of known malicious or undesirable websites is maintained and access to them is blocked. Whitelisting (Altarturi & Anuar, 2020; Federal Communications Commission, 2020; Turner, 2022; Vondersaar, 2020), on the other hand, allows access only to pre-approved websites, ensuring a more controlled browsing experience (Ali et al., 2019; Thangaraj, 2019). These techniques can be combined and customised to suit the specific needs and policies of an organisation or individual user.

Different motives are employed in deploying content filters depending on the organisation using the filtering system (Sartor & Loreggia, 2020; Hidalgo et al., 2019). They could be used to

prevent users from even attempting to access a certain website or webpage to make it unavailable to those who choose to view it. The organisation that wants the filtering will have to decide which technique to use based on its resources, the individuals it can impose it on, and how much resources is required for the implementation. The quantity of allowable errors, whether the filtering should be overt or covert, and its dependability are other factors to take into account (Adam, 2019). The author opined that many groups, institutions, businesses, and nations want to limit Internet access within their borders and physical locations. For instance, businesses may try to increase worker productivity by limiting access to leisure websites; libraries and schools may try to protect children from sexually explicit content or may be required to do so by law and nations may try to regulate the information that their citizens receive in general.

Case Study 1

Case study 1 is an example of a young boy who was exposed to a video of the murder of an airline pilot by ISIS (ITU, 2009). The boy and his mother were in a car on his way to school and the radio station broadcasted the story of the murdered pilot. Innocently, the boy asked the mother questions about this sad news, however, the mother was not prepared to talk about the news. She immediately turned off the radio and they drove in silence. However, this sad news troubled the young boy - the pilot had been burned alive. As soon as they arrived home, he carried out an online search to find out more and attempted to understand what had happened. Of the content presented to him by the search engine was a video showing exactly what had happened; it was posted by a news channel. The young boy revealed that he knew he should stop watching it as soon as the video started but somehow, he could not and he watched the video till the end. It really made him upset; he started having nightmares and became very distressed by the whole experience. Sadly, he refused telling anyone for fear of what their reaction would be and frankly felt that he would be blamed for watching the graphic video.

CBS News (2015), ISIS Video Shows Jordanian Pilot Being Burned to Death, https:// www .cbsnews .com/ video/ isis -video -shows -jordanian -pilot -being -burned -to -death/

2.4.1 The History and Concept behind Web Content Filtering

The evolution of Web content filtering was adopted from the work of Stoll, et al. (2020). Content filtering, an alternative term for censorship is internet content filtering. Internet censorship refers to the limitations placed by governments or organisations on what can be published, accessed, or watched online. It is used to limit or manage the content that users are permitted to view. The main purpose of this is to control content that is sent over the Internet by email, the Web, or other online channels (Voros et al., 2023). Similar filters are frequently used in workplaces to prevent employees from accessing Facebook while at work, but in recent years, there has been a growing movement to impose internet content filtering in order to shield kids from inappropriate content and limit their exposure to sexually explicit content (Ozimek & Förster, 2021). Depending on political, religious, and legislative factors, censorship differs in kind and degree between nations and states. It goes without saying that different entities take censorship to entirely different levels, and they have different goals (Ververis et al., 2021). Certain nations employ censorship as a means of suppressing dissenting opinions; one such example is North Korea, which has complete control over all internet-connected devices (Zittrain et al., 2017). Technologically adept users may frequently overcome content blocks by finding ways to access it. Examples of this include people trying to access Bloomberg, Facebook, and Twitter in China.

For most of online users, web blocking is still seen to be an efficient approach to limit access to malicious or unwanted information. The IP address protocol prevents access to particular IP addresses, DNS filtering and redirection, which has impact on domain name servers, as well as packet filtering, which restricts access to undesirable content; these have been seen as the major approaches for restricting access to the internet.

For the majority of users, blocking is still an efficient way to restrict access to important information. Internet Protocol (IP) address blocking, which prevents access to specific IP addresses, domain name system (DNS) filtering and redirection, which affects domain name servers, and packet filtering, which blocks contentious terms, are the main methods for preventing access to the internet. Websites with copyright issues are most likely going to be blocked by Google. This is done by Google because of the Digital Millennium Copyright Act, and makes it illegal to produce and share technology, services or applications which were originally meant to circumvent restrictions on access to contents that are protected by copyright.

For example, such contents that infringes copyright, belittle politics, and tarnishes the images of the influential in the society is seen to be offensive to Social Networking Sites, and it goes against moral standards. Protecting individuals from unwanted web content is a section of an increasing global movement to enforce content filtering in schools to protect and safeguard children and to reduce web filtering globally to support information freedom against Autocratic leadership.

2.4.2 Benefits of Web Content Filtering

Phishing, viruses, hacking and malware are some of the most serious threats to any organisation; for instance, phishing occurs when hackers obtain sensitive info through deception (tricking employees into logging in to false company's profile to obtain login credentials. This could be disastrous to any organisation resulting in the loss or corruption of sensitive data (login credentials, personally identifiable information (PII), legal documents and accounting information. As a result, content filtering protocols are absolutely necessary. Web content filtering has numerous advantages for both the individual users and for organisations.

1. **Learner Benefits**

The need to protect students' online safety is growing more and more important in the current digital era. Schools are realizing the necessity for effective content-filtering systems to safeguard pupils from potential dangers and distractions as a result of the availability of information online; as a result, web filtering in schools is essential for ensuring that pupils are protected online.

Web filters are a crucial part of any school's technology security plan since they provide several important advantages for educational institutions. Not to mention the advantages it offers both teachers and students directly. It enables smooth instruction on the one hand, and it also makes learning possible and secure on the other.

Every educational establishment should use a school content filter as a useful tool to control pupils' access to online content. It makes it easier for pupils to navigate and addresses preventing inappropriate content in educational settings, which gets rid of dealing with inappropriate internet items. Other than what is forbidden on the school's internet, some web filtering software enables parents to apply other content restrictions. It can restrict or permit access to websites and prevent particular file types.

i. Student Security

One of the most effective methods for ensuring student safety is to use content filtering technologies. Content filtering in schools prevents students from accessing dangerous or otherwise unsuitable and undesirable content. Sites that promote hatred or discrimination, include sexual or violent content, or engage in illegal behaviour are examples of this. Schools can protect children when they use school-provided devices and networks by restricting access to specified websites and material. Furthermore, the protection is not restricted to school hours or premises, as these solutions function on a variety of operating systems and allow parents to participate in their children's online safety. Although content filters cannot avoid more complex techniques, they are good at filtering violent and hurtful content, such as cyberbullying, suicide threats, and detrimental searches about gun violence and school explosions, among other things.

ii. Student and Data Protection

In addition to protecting pupils from unsuitable content, school content screening protects their privacy. Although content filters cannot prevent all sophisticated online threats, they do efficiently prohibit violent and damaging content, such as cyberbullying, suicide threats, and risky searches for violence. Furthermore, content filters can limit access to websites and platforms that are vulnerable to cyberattacks, protecting student data, school devices, and the network. Some content filtering solutions also include location detection tools to assist in the recovery of lost devices. When every student gets access to a school device, the chance of pupils being exposed to hazardous programs such as phishing schemes, viruses, and spam increases, and, because students are more vulnerable to becoming victims due to their digital inexperience and more creative deceit, extra caution is required, because a pupil may inadvertently download harmful software.

iii. Enhancement of Classroom Screen Monitoring

Content filters operate in tandem with classroom screen monitoring systems to make online activities safer and easier for instructors to monitor. While technology cannot replace the full learning experience, it is an important tool for teachers to use in order to improve their teaching methods. The learning experience can be maximized by combining the safety given

by content filters with the opportunity for efficient and engaging learning provided by classroom screen monitoring solutions.

iv. Limits Access to Social Networking Sites

Access to objectionable content, such as pornographic and gambling websites, can be restricted using web filters. Additionally, they aid in keeping kids from downloading spyware or viruses that can harm their computers or data. Cyberbullying and other online threats are less prevalent in schools that utilize Internet filtering software. Access to objectionable content, such as pornographic and gambling websites, can be restricted using web filters. Additionally, they aid in keeping kids from downloading spyware or viruses that can harm their computers or data. Cyberbullying and other online threats are less prevalent in schools that utilize Internet filtering software. Both teachers and students can benefit from preventing students from accessing improper content online. Some social media platforms have objectionable content that should not be viewed by kids or teenagers. These websites can be blocked by a school web filter by preventing access to them using school computers' internet browsers. The school can shield its pupils from content that might be physically or emotionally harmful by blocking these websites.

In conclusion, by using school web filters, schools may increase student security, prevent kids from accessing harmful content, and streamline network maintenance duties. Additionally, web filters have various other benefits, including improved software for screen monitoring in classrooms and improved student focus. In order to protect kids and improve the learning environment, filters are a necessity for any school.

2. Organisational Benefits

i. Preventing malicious websites and objectionable content

By using online traffic and content filtering, employee access may be limited to useful, suitable, and pertinent information while preventing them from viewing hazardous or malicious websites or content. Unrestricted internet access might expose users to offensive, dangerous, or malicious content. This might take the shape of malicious, risky, or pornographic websites or it can involve the development of offensive content for social media, blogs, videos, or other websites.

ii. Blocking Phishing Attacks

One of the greatest hazards to business growth and development is phishing; phishing is a technique for tricking people into giving up sensitive information, such as by pretending to be a firm in order to get login information or seducing workers into sending money to a criminal's bank account (Alanezi, 2021; Alzubaidi, 2021; Bandari, 2023). The bulk of harmful emails can be blocked by spam filters, but some phishing emails can get past perimeter security, especially if they contain links to dangerous websites. Phishing attacks are one of the most common attack mechanisms used by attackers; financial losses, reputational damage, and identity theft are all possible outcomes of successful phishing (Naqvi et al., 2023). By stopping consumers from visiting dangerous websites supplied to them via social media and emails, a web filter offers an extra layer of defence against phishing attacks. Access is denied and the user is redirected to a block screen when a known malicious website is attempted to be accessed (Alanezi, 2021).

iii. Monitoring Internet Access and Blocking Inappropriate Websites

Unrestricted and unhindered access to all web pages may cause a large decline in work efficiency, and workers might lose a staggering amount of time surfing the Internet. For instance, an organisation with about 100 employees would lose about 26,000 hours a year's worth of time if an employee spent an hour every day surfing the web instead of working. A web filter can be used to limit access to websites and webpages like social media sites, and gambling sites, all of which are significant productivity drains (Aktay, 2018; Altarturi & Anuar, 2020). Web filters can also be used to keep an eye on online behaviour (Us, 2022). Employees focus will be on productivity and performance output if they are made aware that the employer monitors their Internet use. Web filters can also be deployed to restrict content that is not suitable for work (NSFW), for instance, pornographic sites. By preventing employees from engaging in illicit online activity (downloading copyright-protected materials from file sharing websites) while at work, web content filters reduce company’s liability (Angelopoulos, 2016; Ververis et al., 2021). Web filters can also restrict activities that consume a lot of bandwidth, such as streaming audio and video (Figueira et al., 2022).

iv. Improve Security Posture

Web content filtering will assist in enhancing security posture, lowering business liability, and enhancing employee productivity. Since the solution is entirely cloud-based, there is no hardware

or software downloads necessary, and it can be put into use right away. By categorizing websites into different pre-defined categories, the technology makes it simple for organisations to ban particular kinds of content. With the use of cloud-based lookup and the database's more than 500 million URL categories, it is feasible to guarantee extremely accurate content filtering while avoiding the over-blocking of valuable content (Ali et al., 2019). The solution is able to examine all web traffic, even that from encrypted sites. The system safeguards workers both on and off the network and enables policies to be defined for the entire workforce, groups, or individuals. The content filtering measures can be implemented uniformly to all devices used by employees and will function whether the user is travelling or on-site. A full reporting suite is available to administrators, with different preloaded reports and room for customisation, as well as options for scheduling reports and real-time report viewing. So, internet filters could help organisations if they want to strengthen their security posture, save bandwidth, lower legal liability, block NSFW content, and increase productivity.

v. Protection against Exploit Kits

The most typical attack vector for distributing malware is email spam, and while though exploit kit threats are not nearly as serious as they were in 2015 and 2016, they still provide a challenge for companies. Software toolkits known as ‘exploit kits’ are used to distribute malware widely by automatically infecting users' machines through websites on the Internet (Nikolaev et al., 2015). Web-based applications known as exploit kits are installed onto websites that are under the control of cybercriminals, either their own websites or ones that have been taken over. Code in exploit kits can take advantage of flaws in plugins, browser extensions, and online browsers. A user's browser vulnerability is exploited and malware is downloaded when users visits a malicious website that encloses an exploit kit. It is now even more difficult to infect PCs with malware using exploit kits, and so, many threat actors have switched to new attack vectors as browsers are now more secure and flash drives are gradually being phased out. Some exploit kits, however, are still operational and a hazard. Numerous exploits for known vulnerabilities are included in the exploit kits now in use (Chhibbar, 2022). Though occasionally zero-day vulnerabilities are released, the majority of the vulnerabilities is outdated and has patches that have been available for quite some time now (Etuh et al., 2021; Kannelønning & Katsikas, 2023). Businesses can improve security by using a web filter, but updating browsers and plugins and using a premium antivirus solution will offer an adequate protection.

vi. Increased Employee Productivity

Social networking sites are a well-known productivity drain because it distracts employees; it can take up a lot of valuable work time, and may lead to lower productivity. Limiting access has shown to increase productivity significantly (Narwal, 2018; Adam, 2019). However, some organisations require social media expertise on a daily basis. Various websites (online shopping and streaming sites) have become productivity sinks in these industries. Employees should be stopped from watching movies when they should be working; it is critical to consider other potential threats, such as downloading suspicious files, opening suspicious emails, and responding to unverified contacts (Jain et al., 2021; Santiago, 2021; Vivekanandam & Midhunchakkaravarthy, 2022).

vii. Reduced Company Liability

An organisation that actively monitors its employees' internet usage is better positioned to mitigate internet-related challenges. Liabilities may result from knowingly or unknowingly sharing offensive content, as well as posting biased or vulgar content, engaging in cyberbullying, or the downloading of copyright-protected content. With the world becoming more radicalized, it is critical to protect the organisation and the brand they represent at all times by carefully monitoring the information sent out by the employers and the employees (Angelopoulos, 2016; NCC, 2018; Umar, 2020).

viii. Network Bandwidth Efficiency

Internet usage that is not related to work consumes a significant amount of network bandwidth (Figueira et al., 2022). An organisation can improve network bandwidth efficiency and achieve faster connections by restricting access to these sites, and by educating their team about the benefits of a consistently fast network or by imposing bandwidth limits on video streaming sites such as YouTube. Furthermore, an organisation can try to impress upon the employees the importance of maintaining available bandwidth for business needs, but it may be more practical and necessary, to set limits on some sites, particularly streaming services such as YouTube, Spotify, and others that require more significant bandwidth (Ahmed et al., 2018).

Case study 2

This case study exemplifies a teenage girl who was being sent objectionable pictures from a man on Instagram. The man shared his nude photos and also asked the girl to send nude pictures to him. However, the young girl did not consent to this, but rather blocked him, and further reported the man to Instagram. The girl talked to some of her friends about her experience with the man, perhaps, the same thing may have happened to them, and it had. Having done all that her senses deemed right, she did not inform her parents for fear of what their reactions would be, such as banning her from using Instagram, which was not an option for her. She revealed that all her friends posted content and shared news, discussed what had happened at school, engaged in gossiping, and made their social arrangements. The issue is that the girl being on Instagram had done nothing wrong; rather, the adult sending the photos was the one who had behaved inappropriately. It is a genuine and an understandable reaction for parents to protect their young people but certainly it is not right to reprimand the child for someone else's wrongs. This case study presumed that most or all of the activities the girl was performing on Instagram was undeniably fine. Parents should consider their reaction when they are told by their children about an issue or a problem that they have encountered or encountering online. Parents should also give listening ears and provide support to their children (ITU, 2009).

2.4.3 Web Content Filtering Deployment

A Web Filter can be deployed at different levels (Figure 2.9).

i. **At National/Country Level**

A country or a nation may deploy internet filters between the National Internet Backbone and the country's network; some countries (China and Saudi Arabia) have deployed and implemented web filters at the National Level; the level of filtering is fully determined by the country's government and her policies, hence, citizens do not have control over what is filtered (Chhibbar, 2022).

ii. At Internet Service Provider (ISP)

The Web Filter is deployed and implemented at the ISP Gateway and gives filtered web content to all its clients; for instance, telecommunication companies may filter web content at this level, and this will affect all subscribers of the communication network (Johanna, 2019; On et al., 2019; Ververis et al., 2021). For instance the Nigerian Government banned Twitter on Friday, June 11th, 2021, for violating what it termed 'abusive behaviour' standards by removing a tweet from the then Nigerian President Muhammadu Buhari, when he tweeted, "Many of those misbehaving today are too young to remember the devastation and loss of life that occurred during the Nigerian Civil War, those who have been through the war will treat them in the language that they understand." President Buhari's Twitter account was suspended for 12 hours and was instructed to delete the tweet. Following the government's ban on Twitter, the then Attorney-General, Abubakar Malami directed the Ministry of Justice to prosecute anyone who contravened the prohibition (Malami, 2020).

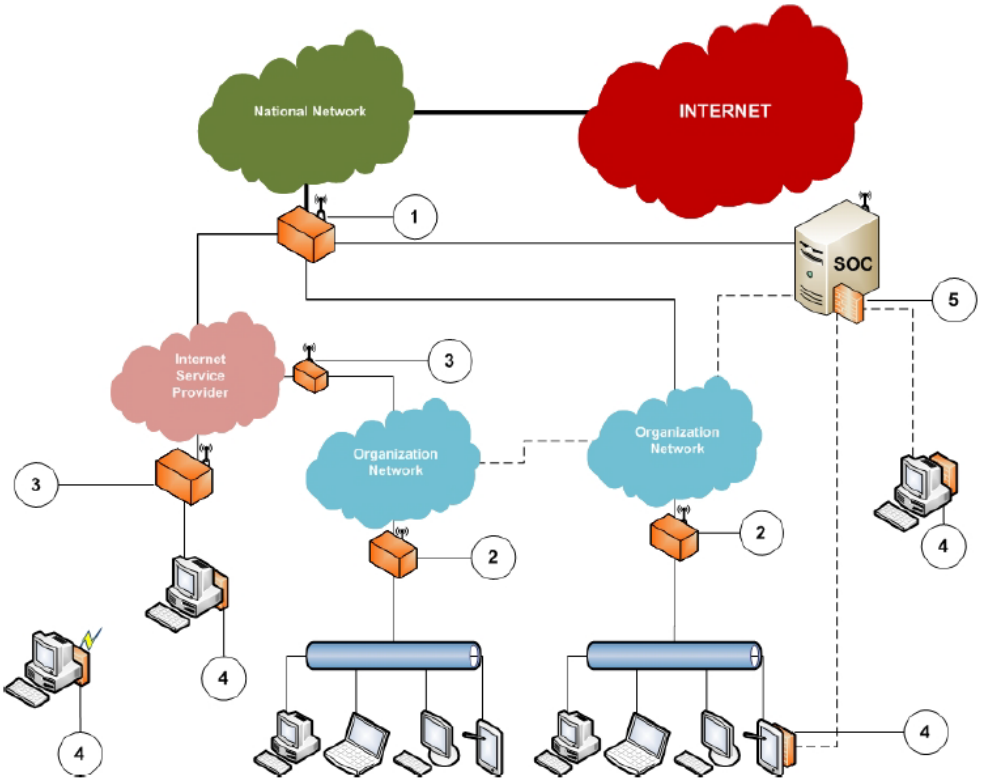


Figure 2.9: Web Filtering Deployment (Banday & Shah, 2010)

iii. At Organisational Level

Filtering at the organisational level is quite different from the nation/country's level. At the organisation's level, the Filter is deployed and implemented between the Organisation's Network and the Internet Gateway, and so all users (all employees) of this Gateway Server are provided filtered Internet Content. The KU Gateway installed at http://192.168.81.251:8090/corporate/servlet/CyberoamHTTPClient is an example of Organisational Level Filtering (Banday, 2014). The installed web filter is customizable (name, logo, duration of use, etc.,) by the organisations web/network Administrators based on the organisational practice and policy regarding what is deemed appropriate or otherwise by the organisation.

iv. At Individual Level

At this level of deployment, the Filter is deployed and implemented at the owner's local computer or workstation. This filtering system may be part of antivirus software, a firewall, an Anti-malware or through other similar systems such as parental control, content advisor, among others. For instance, a child's system may have a filter deployed.

v. At Third-Party

At this level, a web filtering service may be provided by a trusted third party (an individual or an organisation) or software vendor through its Security Operation Centers (SOCs). Here, the client sends their network web traffic through these SOCs by using a proxy server, such as Websense and ScanSafe. Although, third party services could offer filtering at any level for an organisation, but the only limiting factor is that a third party service is only available for small and medium organisations.

2.4.4 How Web Filtering Works

Different people respond differently to different content filters; content filters are available for Internet firewalls as hardware or software; supporting company regulations and cybersecurity in relation to the use of company data systems is the advantage of both features. For example, content filters will obstruct malicious websites (cybersecurity) and knitting social networking sites, according to workplace policy (Federal Communications Commission, 2020; Odiaga et al., 2020). By removing potential risks and the chance that the health of the workforce or systems

will be jeopardized, this aids businesses in providing a safe and orderly work environment for their employees and achieving long-term organisational objectives (Figure 2.10).

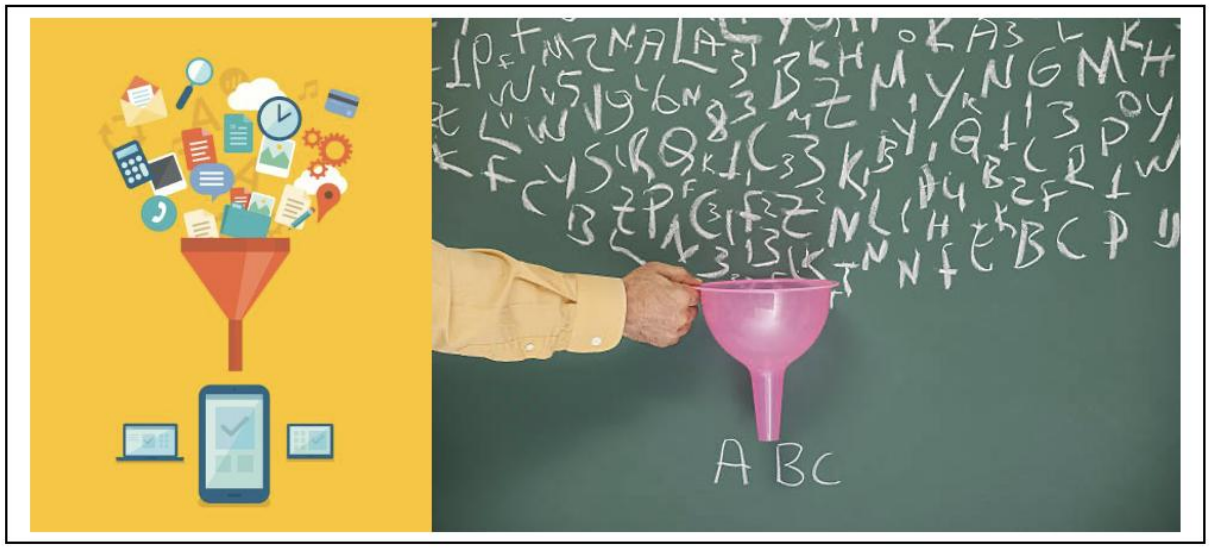


Figure 2.10: How filters work (https://www.gettyimages.com/)

2.4.5 Approaches to Web Content Filtering

There are a few approaches to web filtering, several approaches have been proposed and developed by researchers, some of which are successful in accurately blocking problematic web content (Altarturi & Anuar, 2020; Ali et al., 2021). The following approaches describe web content filtering:

i. Uniform Resource Locator (URL)-based
ii. Keyword-based filtering
iii. Dynamic filtering
iv. Content-based filtering and
v. Image/video-based filtering

i. Uniform Resource Locator (URL)-based

This method approves or disapproves access to web content by comparing the web page's domain (and IP address equivalent) that was requested and the domains kept in a list (Ali et al., 2021). This list contains either the blacklist that comprises the domains of questionable websites to restrict, or the white list contains domains of permitted websites to allow. Lists of both types must be kept up-to-date. It is important to note that blacklists are commonly used in web filtering

systems that involve domain blocking. A good advantage of this approach is that it operates quickly and effectively, which is ideal for any web-filtering system. The system may be able to determine the kind of a web page's content during classification by using sophisticated content analysis algorithms. The domain of the page may be added to the blacklist if the system decides that the content is offensive. The system can swiftly determine whether to filter a user by comparing the URL when they attempt to access the web page; if the content analysis is accurate, dynamic updating of the blacklist is quick and efficient while maintaining accuracy.

This method, however, can only identify the websites on the list and calls for the implementation of a URL list. Additionally, the system's accuracy would rapidly decline if the list is not routinely updated due to the quick proliferation of new websites. Most domain blocking web filtering solutions actively search for problematic websites that could be added to the black-list using human review teams. Table 2.1 shows the examples of some black-listed URLs.

Table 2.1: Black-listed URLs (for research purposes only)

S/N	Domain Name	IP Address
1	https://www.penismedical.com	173.255.194.134
2	https://www.altpenis.com/	72.32.132.134
3	https://www.allabout-penis-enlargement.com	38.238.78.168
4	https://www.omegle.com	104.23.141.25
5	https://www.toomics.com	65.49.30.176
6	https://www.reddit.com	151.101.193.140
7	https://www.tumblr.com	192.0.77.40
8	https://www.monkey.cool	52.11.27.227
9	https://www.twitter.com	104.244.42.65
10	https://www.tinder.com	52.84.150.39
11	https://www.chatroulette.com	147.75.75.179
12	https://www.chan.com	45.33.2.79
13	https://www.ask.fm.com	193.138.77.140

ii. Keyword-based Filtering

The keyword matching method of web content analysis is the most basic, and this logically straightforward method restricts access to URLs which has components of objectionable words and phrases; each word or phrase detected on a retrieved website is compared against a keyword dictionary of prohibited terms and phrases, and when the number of matches hits a certain threshold, blocking happens (Ali et al., 2019). It is simple to determine whether a web page includes potentially dangerous content using this quick content analysis approach. However, this approach has some limitations, such as over-blocking, which is the act of blocking numerous websites that do not have undesirable content; because it matches words (or phrases) like 'sex', 'vagina' and 'breast' when filtering material, it may unintentionally ban websites that discuss sexually transmitted disease, sex education, breast feeding or breast tenderness, vagina candidiasis, among others (Altarturi & Anuar, 2020). A web-filtering system's performance is usually jeopardized by the high over-blocking rate, which is frequently undesired, though the list of offensive terms and phrases does not necessarily need to be updated frequently. Appendix G spells out the list of black-listed words and their corresponding effects. A web filtering system can use this approaches to decide whether to use a more accurate web content analysis method, which typically requires more processing time.

iii. Dynamic Filtering

In this type of filtering approach, web contents are dynamically updated, this approach reduces the under blocking limitation offered by the URL-based filtering approach. Kashmar et al. (2020) supported that dynamic filtering is beneficial due to the ability of inspecting the different components of a website/webpage for categorization which includes the metadata, images, and textual content. However, Shu et al. (2020) had a contrary idea about dynamically created content on social networking sites and other platforms such as chats, https/http, and SSL creates some technical and practical inadequacies for the filtering technology, which may lead to inadequate performance. A further addition to dynamic filtering approach is supported by Darer (2020) that most commercial filtering techniques only have to use dynamic filtering approach to enhance more efficient filtering techniques.

iv. Content-based filtering

The content of any given webpage is further examined for confirmation of its class in order to address the restriction of URL-based blocking/filtering; every web page has HTML tags that are

filled with meta-information about the page. These can be used to determine whether or not the content is inappropriate (Ali et al., 2021). Moreover, a website can be categorized into groups like news, entertainment, and education. Websites are filtered using the content of websites and/or each web page. This approach is used in literature for a variety of purposes, including information retrieval, structuring web information sources, and search queries. This method was researched in literature, which also provided a thorough overview of web mining in general; following that, literature began to examine the web filtering area in more detail by addressing its algorithms, advantages, disadvantages, and challenges (Altarturi & Anuar, 2020; Ali et al., 2021).

v. Image-based filtering

According to Vivekanandam & Midhunchakkaravarthy (2022), image-based filtering is an dynamic filtering research area because of the increasing volume of images and multimedia components available on the web. Several commercially available software classifies web content as either pornographic or not, and this is achieved using only textual content on the website. Nonetheless, Desai et al. (2020) opposed text-based filtering as not effective as websites and webpages also contain images with little textual content. Reacting to this, Kaur & Singh (2017) stated that Image filtering approach that is majorly based on skin detection is an evolving technology which has a high degree of accuracy, however, they have slow performance, which makes this technique impracticable in real-world systems. However, this image-based filtering is out of the scope of this research work. Table 2.2 shows the summary of content filtering techniques.

Table 2.2: Summary of Content Filtering Techniques

Internet Content Blocking Techniques					
	IP and Protocol-Based Blocking	**Deep Packet Inspection-Based Blocking**	**URL-Based Blocking**	**Platform-Based Blocking (especially search engines)**	**DNS-Based Blocking**
Overview	A device is inserted in the network that blocks based on IP address and/or application (e.g., VPN)	A device is inserted in the network that blocks based on keywords and/or other content (filename, for example)	A device is inserted in the network that intercepts web requests and looks up URLs against a block list	Working with application providers (such as search engines), content is modified according to local requirements	At the network or ISP level, DNS traffic is funneled to a modified DNS server that can block lookups of certain domain names
Is it effective?	Because IP addresses are easily changed and content easily moved, **this technique works poorly.** This only works well when the information publisher is not actively working to evade the block.	Where the blocked information is easily characterized, this is very effective. For general blocking (e.g., "block adult content") or in the face of encryption, **the technique is very ineffective**	This is a common technique that **works well when blocking access to entire categories** of information. New pages and smaller sites slip through easily, as do encrypted web servers.	Because there is no monopoly in search engines (for example) and consumer preferences are constantly changing, **this type of blocking is largely cosmetic and works poorly.**	DNS blocking is easily evaded both by content publishers and end users. DNS blocking is only effective when each name has a very small amount of content, and all that content should be blocked. **Technical challenges, over-blocking, and ease of evasion make this an ineffective technique.**
Who is affected?	Anyone who is "behind" the device is affected.	Anyone who is "behind" the device is affected.	Users "behind" the device, and for whom the device can intercept and evaluate web traffic.	Users of the search engine which has installed the block	Users of the modified DNS server. This can be enforced at the network or service provider level.
How specific is it?	**Affects all content on a server, whether illegal or not.** This works even when the data are encrypted.	**Affects only content which matches blocking rules.** Requires proxies to work with encrypted web pages.	**Affects individual web pages and web elements.** Requires proxies to work with encrypted web pages.	**Affects individual web pages and elements.** Usually done at the individual URL level.	**Affects all content served by a domain name, whether illegal or not.** Cannot be effectively used to distribute content.
What type of technique is this?	Blocks content	Blocks content	Blocks content	Discourages and frustrates access	Discourages and frustrates access
How much collateral damage is caused?	Any targeting of larger servers has a huge false positive rate, blocking both illegal and legal content.	Depending on the quality of the blocking rules, the false positive rate can range from very low to quite high. Writing good rules is difficult.	Most URL filtering is based on commercial services that categorize traffic. For mainstream blocks, this can be quite specific, but for special purpose blocks, the error rate is quite high.	The false positive rate is considered to be low, because each page block is requested individually. The problem of non-legitimate requests causes some inappropriate information to be blockage.	Any targeting of domain names used by larger servers has a huge false positive rate, blocking both illegal and legal content. Ineffective when CDNs are used (or causes an extremely high level of false positives).
What are common ways to evade it?	Publishers can change IP addresses, migrate content, or use Content Delivery Networks (CDNs) to evade. VPN users evade by hiding IP addresses.	Multiple layers of encryption effectively evade this type of blocking. When the filtering rules are poorly written, small changes in text can easily bypass blocks.	Multiple layers of encryption effectively evade this type of blocking. Use of non-standard application layer is often an effective evasion technique.	Users can choose alternative platforms, such as a different search engine, very easily.	Users can avoid using DNS lookups using local facilities, or can send their queries to an un-modified public server (typically though a VPN).
Are there side-effects or technical issues?	Maintaining long IP address lists is difficult and error-prone, and requires significant resources. Network devices doing this type of blocking are typically speedy, so performance issues are not common.	Content-aware filtering has significant performance costs and is not practical in many environments (without enormous resources). When proxies are used, security can be severely compromised.	URL filtering can cause performance problems, decreasing overall speed and reliability. When proxies are used, security can be severely compromised.	Many search engines report on "suppressed" information, which itself creates a trail to the content.	DNS security is compromised when a modified server is deployed.

2.5 Current and Emerging Issues of Online Threats and Security

This section presents a systematic and in-depth exploration of current and emerging issues of online threats and security. It x-rays the conventional cyber threats that affect most social network users and the current cyber threats that are currently prevalent. The main focus is to provide insight into social network security and protection. It discusses all the likely aspects of online social networks and related issues. It also sheds more light on the established open challenges and concerns that need to be addressed in order to enhance the credibility of online social networking sites.

Over the past 10 years, social networking sites have expanded, and these forums have attracted a lot of media interest (Nawaz et al., 2023). Through the use of online social networks, people are now actively engaged in virtual communities and communicate with friends and family who share common interests by exchanging information, ideas, and other kinds of expression. However, as social networking sites expands, the social sphere also became increasingly commercialized, creating concerns about user security and privacy (Abba & Hassan, 2022). Hence, social networking sites are a prime target to attackers due to the abundance of personal information that is available on their platforms (Rathore et al., 2017). Meanwhile, social networkers often neglect the significance of protecting the personal identifiable information (PII) that is kept on social networking sites as they are often viewed as personal communication tools (Jain et al., 2021). However, this assertion was refuted by Zeebaree et al. (2020) that users take information protection for granted as increasing use of social networking sites (SNS) by hackers cannot be disregarded. In the words of Nawaz et al. (2023), social networkers frequently provide a variety of details, such as name, age, current job, gender, address, and photos, as may be needed by the social platform (Figure 2.11).

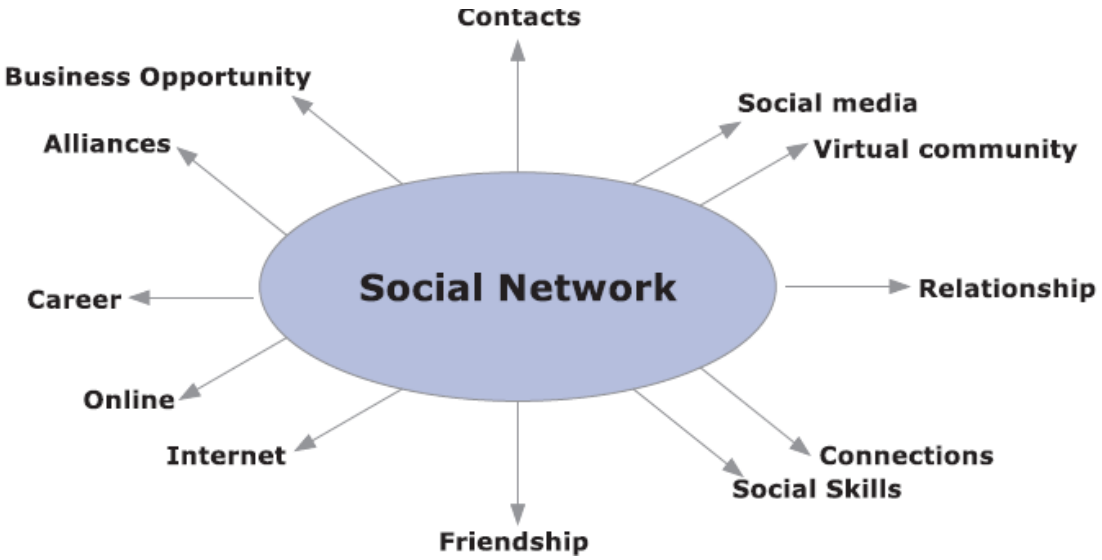


Figure 2.11: Fundamental Concept of Social Networking Sites (Nawaz et al., 2023)

2.5.1 Online Social Network and Media Threats

Traditional/Narrative literature Review technique was employed to broadly identify and review already published related studies. The approach typically amasses a wide range of related subjects to address research objective one (section 1.4) of this study and reporting the review findings in this section.

Different technological gadgets are providing a seamless platform for users to be connected to the Internet and to perform different tasks; this has extended interactions between humans and machines through the Internet. These are some of the online attacks which users have had to experience from the inception of the concept of social networking sites (SNSs). The threats have been categorized into three, name; conventional, modern, and targeted threats (Figure 2.12).

Conventional threats refer to threats that have been experienced by online users' right from the introduction of social networking sites. Modern threats refer to online threats that deploy cutting-edge techniques and tools to compromise the accounts of users, while targeted threats are attacks that are focused or targeted on some particular individual(s) which may be perpetrated by any attacker for varied personal reasons.

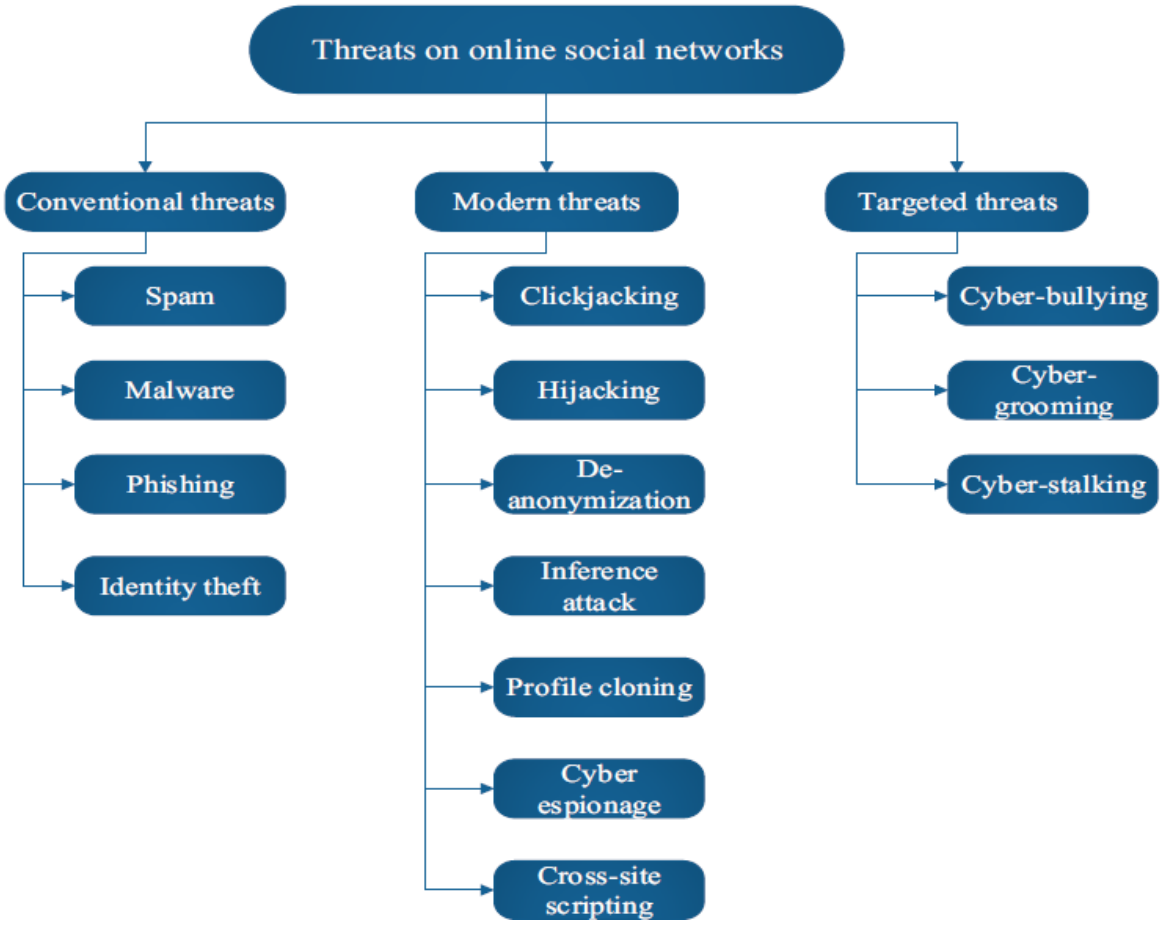


Figure 2.12: Classification of Online Threats (Jain et al., 2021)

Considering the online safety of social networkers, a literature review reported that there is the need for online security and safety to be taught to social networkers Macaulay et al. (2019). For instance, the study conducted by Veli (2019) whose main focus was on 271 young online users between the ages of 7 and 11 years which examined how this group perform their online activities. The report emphasized that e-safety policies and procedures are not updated as technology evolves and established that extra resources are required to educate online users on the need to observe safe practices while surfing the Internet. To further buttress this, the researcher, Cohen (2019) identified the need of supporting online users to develop online safety skills. According to the researcher, many research works conducted across the globe employed online gamified activities, stories, and multimedia to teach online safety strategies to students, focusing on several levels of online safety, such as avoiding identity theft, preventing cybercrimes, protection from hackers, from cyber-bullying, and the protection of personal data. In a general term, the outcomes from the reports of these researches were quite encouraging.

1. Conventional threats

i. Spam Attack

Spam is the word that describes any unsolicited or unwanted electronic messages (emails) (Rapacz et al., 2021; Kaddoura et al., 2022). Although, it is generally believed that email is the popular way for the spread of such messages, social networking platforms too are further involved in the spread of spam messages Fatichah et al. (2021). The information of online users can simply be gotten from the websites of organisations (subscribers' list), blogs, fake websites, newsgroup, and many more (Zeebaree et al., 2020). More so, users are easily convinced to read spam messages and unfortunately, believe that it is genuine. Although, majority of the spam messages are commercial advertisements; however, users' sensitive information can be collected by this means or they may be prone to malware, viruses, or scams (Herath et al., 2022).

ii. Malware Attack

The term Malware was coined from the words **mal**icious soft**ware**, which refers to harmful software which is intentionally meant to infect or grant access to a computer system without the consent of the user (Perera, 2021). According to Faruk et al. (2021), with fast-pace of emerging technological advancement, online security is now a critical issue due to a steady increase in malware activity, which poses a serious cyber threat to the security and safety of both PCs and users. The protection of data from fraudulent attempts records as one of the most pressing concerns for end users. Malware, therefore, is a collection of malicious scripts, programming code, active content, or intrusive software that is solely intended to destroy computer systems and programs, as well as mobile and web applications (Alzubaidi, 2021; Nicolaidou & Venizelou, 2020). An attacker has different methods to spread the attack of malware and infect devices and networks Herath et al. (2022), for instance, malware may be installed or unknowingly downloaded by merely clicking an unsuspected malicious link or the client might be diverted to a deceiving website which may lead to identify theft or some attackers might inject some malicious scripts in websites and any user that visits the websites will activate the script on the system which may again lead to identity theft of sensitive information. According to Jain et al. (2021), the approach adopted by malware is by using online social network's structure to spread (the number of vertices, number of edges, average shortest path, and longest path).

iii. Phishing

This type of attack is a social engineering attack in which the attacker trickily obtains sensitive and confidential information from the user, such as usernames, passwords, credit card information, etc., through false websites and emails that appear to be legitimate (Bandari, 2023). Most times, an attacker can impersonate the real or original user and may use the real owner's identity to send fictitious messages to other online users through any of the known social networking platforms which may contain malicious scripts, which may also redirect the authentic to fake or fictitious websites, and may also request for personal details (Ozbay & Alatas, 2020). In relation to Social Networking Sites, an attacker tends to attract the user to a fictitious website where the phishing attack could be executed. In order for this to be accomplished, different social engineering methodologies may be deployed by the attacker. For example, messages like, 'you have just won a prize', or 'Congratulations, click the link below to claim your gift', or 'Admin, kindly assist in resetting my password.' Any attempt by the user to click as directed will direct the user to a fake website where the exploitation will take place. According to Kumar et al. (2022), phishing and attacks on social network accounts of information systems are major industry concerns and that unauthorised users who have access to information from social networking sites may use it for illegal purposes, including hacking, spoofing, and phishing, among other things, endangering the safety and security of users of online social networks. The targets of phishing attacks have shifted from individuals to businesses, according to Phishlabs' 2018 research on phishing trends. Phishers presently use free SSL certificates to make matters worse. As of right now, HTTPS is used by around 50% of all phishing websites, making it one of the key indicators of a website's legitimacy (Alanezi, 2021).

iv. Identity Theft

In identity theft, the attacker uses another person's identity (social security number, mobile phone number, and address) to commit crimes without permission (Herath et al., 2022). It is easy for an attacker to gain access to a victim's friend list and demand personal information from them using various social engineering techniques. Since the attacker is impersonating a real user, the identity could be used in any way imaginable, which could have catastrophic consequences for legitimate online users (Jain et al., 2021).

2. Modern Attacks

i. Cross-Site Script (XSS)

XSS refers to a network of human intruders who insert malicious script codes into dynamic web pages (Jain et al., 2021). When these web pages are accessed by a user, the browser automatically downloads these malicious codes that were embedded in the pages. Once interpreted, the malicious script codes allows the attacker to get around security restrictions in the Document Object Model (DOM), steal cookies, and alter account settings. In order to accomplish an illegal special objective and steal user information, web applications and web-mailing worms propagated a number of malicious acts (Thajeel et al., 2023).

ii. Hijacking

For an online crime to be committed, the attacker must take control of the victim's account (Jain et al., 2021); and because credentials can be stolen via phishing, online platforms that do not implement the multifactor authentication (MFA) and users' accounts with weak passwords are more vulnerable to this type of attack; in essence, online users lack a secondary line of protection if they do not have multifactor authentication (Vivekanandam & Midhunchakkaravarthy, 2022). Once an account is hijacked, the hijacker is able to send messages, publish dangerous links, and change account information, thereby harming the victim's reputation. According to Ogundele et al. (2020), the most common sort of attack is session hijacking, because of the ease of accessing the session it is frequently favoured by attackers. The attacker hijacks the session by masquerading as the real user when a user is logged in or about to log in to the system and has established a connection with the server (Figure 2.13).

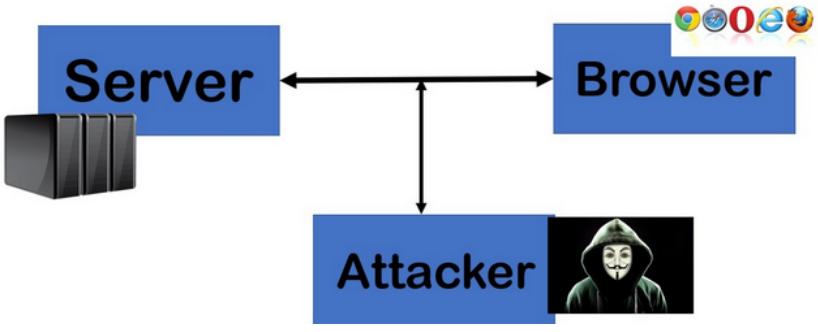


Figure 2.13: Session Hijacking Attack (Jain et al., 2021)

iii. Profile Cloning Attack

This is an attack technique that copies a user's profile, thereby duplicating the victim's profile. The attacker may use this duplicate profile on the same or a different social networking site to build trust over time with the genuine user's friends (Jain et al., 2021). Once a link is established, the attacker deceives the victim's friends into believing the deceptive profile is real in order to obtain private information not displayed in their public profiles. This method can also be used to commit other online crimes such as cyberstalking, cyberbullying, and extortion (Jain et al., 2021; Perera, 2021; Vivekanandam & Midhunchakkaravarthy, 2022).

iv. Click-jacking

Clickjacking, otherwise known as User Interface redress assault and it is an attacking technique in which an intruder tricks an online user into clicking on a page that is not what is intended (Nawaz et al., 2023). To carry out this attack, the attacker takes advantage of a browser flaw, by placing another page as a transparent overlay over the page that the user wishes to visit (Abba & Hassan, 2022). Clickjacking has two recognized variations: likejacking and cursorjacking (Jain et al., 2021). The front layer depicts the material that can be used to entice the client. The client taps the like button after touching the content; the more people who like the post, the more likely it is to go viral. Cursorjacking occurs when an attacker substitutes a modified cursor image for the actual cursor, and the actual pointer is moved from the current mouse position, to trick a consumer into clicking on the malicious site by cleverly placing website elements (Jain et al., 2021; Rathore et al., 2017).

3. Targeted Threats

i. Cyberbullying

Bullying or harassing someone via electronic communication tools such as emails, chat rooms, phone calls, and social media sites is referred to as cyberbullying (Herath et al., 2022). In contrast to traditional bullying, cyberbullying is a continuous process (Nash, 2021). The attacker regularly sends intimidating and threatening messages, for example, making sexual remarks, spreading rumours, and intermittently uploading embarrassing images and/or videos to harass a person (Jain et al., 2021). The attacker may also leak sensitive information about the victim, which may be either humiliating or embarrassing. Knowing the attacker's tone through text

messages, instant conversations, emails, or other forms is extremely challenging; however, the patterns that appear repeatedly in these emails, texts, and online posts are rarely accidental (Saliu et al., 2022).

ii. Cyberstalking

This is an unhealthy practice of following someone online, either through emails, or other types of electronic interaction that has the tendency of causing the victim to be concerned about their safety and disturbs their mental peace (Hasselt & Bourke, 2018; Dhillon & Smith, 2019). Cyberstalking which occurs when a person is persistently contacted online in a way that makes them feel frightened or uncomfortable, is at the extreme and most serious end of the spectrum. Although there is no universally agreed-upon definition of cyberstalking, it is generally agreed that stalkers pursue their targets online or through some other form of telecommunications, example, through emails and text messages (Krasnova, 2022). The right to privacy of an individual is violated by this act; and the attacker keeps track of the victims' private or sensitive information and uses it to threaten them continuously and persistently all the time. Behaviour such as this rouses a form of anxiety, fear, or disturbance in the victim and causes the need for the victim to be extremely worried for his safety. Most online users provide personal information in their social networking profiles, such as phone number, home address, neighbourhood, location-based information, schedule, and other personally identifiable information (Hasselt & Bourke, 2018). Offensive name-calling, intended embarrassment, stalking, physical threats, harassment over a continuous time, and sexual harassment are the six unique behaviours used in the Online Pew Research Center poll to quantify online harassment (Sasaki & Hobbs, 2019).

iii. Cyber Grooming

The act of establishing a close and emotional bond with the victim through cyber grooming is done with the goal of coercing sexual assault (Veli, 2019). The main idea behind cyber grooming is to win over a child's trust so that more personal and sensitive information may be extracted from the child; the information is frequently pornographic in nature and is obtained through sexual discussions, images, and films, giving the attacker the upper hand in threatening and

extorting the youngster (Jain et al., 2021). Since the youngsters have been lured in with the intention of cyber grooming, attackers frequently approach teenagers or children using a false identity (Vivekanandam & Midhunchakkaravarthy, 2022); consequently, the victim may also unintentionally start the grooming process when lucrative offers, such as money or other valuables are promised or presented in exchange for contact information or pictures of themselves are being offered.

2.6 Current Security Control Measures on Social Networking Sites

Users of ICT gadgets now have access to a huge variety of web information, including undesirable content, and young online users are put at risk by these undesirable contents (Ali et al., 2021). Several studies (Hasselt & Bourke, 2018; Baldry et al., 2019; Maserumule, 2020; Altarturi & Anuar, 2020) show the need to study cyber parental controls as studies on the approaches, methodologies, and datasets in this area are nonetheless few. This issue has therefore prompted researchers in academia and business to create parental control software for web content filtering (Ali et al., 2021). The advancement of online technology in the twenty-first century has resulted in limitless and timeless communication, to communicate through social media platforms (Ayubi et al., 2020). To describe the communication in question, Ayubi et al., (2020) stated that terms such as social media, social networks, social network sites, social networking sites, and online social networking are used interchangeably. By explaining these concepts, the correct use of these terms will be ensured.

Five of 34 related studies reviewed were more concerned about a high users vulnerability to online activities affirming cyber insecurity as a serious issue in social networking (Jain et al., 2021), particularly with attackers developing fresh strategies and entry points to gain unauthorized access into the accounts of users every day to gather and retrieve personal data (Nawalagatti et al., 2022). This study discovered that different control measures have been adopted to prevent cyber-attacks on SNS, and researchers have also developed a number of technical solutions to safeguard online users, but information security is still insufficient considering the multiple incident reports attesting to the rise in security breach frequency (Etuh et al., 2021; Kashmar et al., 2022; Ogundele et al., 2020; Okesola, 2014 Vivekanandam & Midhunchakkaravarthy, 2022). Amongst the most common solutions is users' credentials

authentication, such as username and password (PIN), biometric authentication (facial, voice or fingerprint recognition). It was also gathered that data protection could be solely (through the use of passwords) or multi factor authentication, which combines a password with a PIN or other biometric information. Sole data protection or Single authentication is characterized with weak passwords that may be easily exploited. These limitations have therefore inclined many researchers to propose more effective security mechanisms (such as Biometrics, Intrusion Detection System, content filtering, etc.,) to curb cyber-attacks (Petru-cristian, 2023; Zwilling et al., 2022). Figure 2.14 shows some of the best practice for social media online security.

Regrettably, several studies (Abdallah et al., 2020; Egenti et al., 2023; Garba et al., 2022; reported that inadequate users' awareness on security is the major control issue in SNSs. Potgieter, (2019) even submitted that the successful implementation of cyber-attacks could be attributed to users' ignorance to cyberspace risks, and therefore insisted that users should create an awareness culture before entering online ecosystem. Consequently, educational establishments are now involving in raising peoples' understanding of cyber security issues for users to routinely interact with the communication medium on online security issues. Participants in the teaching-learning process are also required to receive proper training since ignorance can lead to unsafe online behaviours that increase the vulnerability to cyber-attacks (Bottyán, 2023).

Furnell and Vasileiou, (2017), opined that because students are less aware of security threats, the potential harms, and preventive measures; therefore, their ignorance can be linked to a lack of security education. This is because education and security knowledge are often accorded very little attention. Similarly, many organisations do not take any action at all, while others rely on universal solutions that are unlikely to have an equal impact on every employee. Hence, Furnell and Vasileiou, (2017) insisted that personalized awareness, training, and education must be designed to target recipients more precisely by taking into consideration variables - the individual's role, perception of security, learning style, and past knowledge.

While academia have come up with creative approaches to address the security measures related to social media security, these approaches are seen to lack practicality and are more difficult to implement in the real world. To stay up with technological advancements, security vulnerabilities in social networks must be often reviewed and regularly too. The majority of this analysis makes it clear that organisations pose significant risks to online security and safety, and

therefore, companies should take the necessary precautions to prevent cybercrime, and individuals should safeguard their personal data to prevent exploitations of any sort. Vulnerability in the security measures put in place on SNSs makes these applications insecure, as many users share sensitive personal information. Since cyberspace is increasingly becoming a major space for criminal activity, thorough international cooperation is required to fight the ever-growing threat of social network security and social media cyber-attacks.

Figure 2.14: Current online security best practices for social media usage

2.5.2 Solutions for Various Online Threats

Numerous experts from both academia and business are working nonstop to develop answers to the aforementioned social media risks. They have put out a number of ideas and strategies to counter these dangers (Jain et al., 2021). This section explains the numerous techniques and strategies used by social networking sites' security that have been researched by various researchers. The solutions have been divided into two categories: those provided by social network operators and those provided by academic institutions (Figure 2.15).

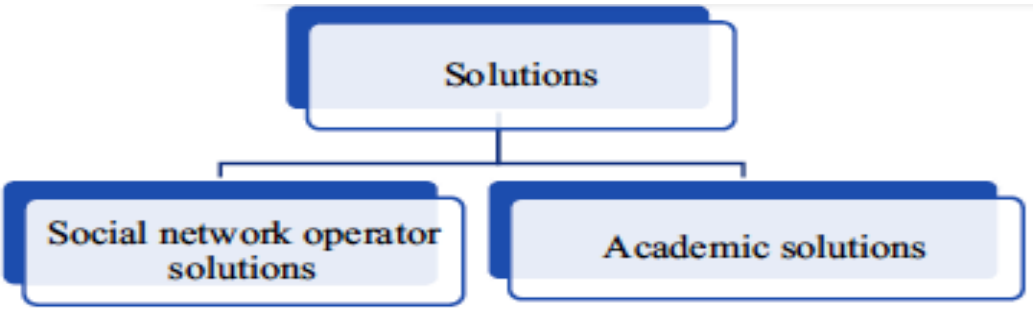


Figure 2.15: Classifications of Threat Solutions (Jain et al., 2021)

1. Threat Solutions by Social Network Operators

i. Authentication Procedure

Several online social networks use different authentication processes, like CAPTCHA, two-factor authentication (2FA), multi-factor authentication (MFA), photos-of-friend identification, etc., to ascertain that only a real user and not a social-bot is logging in or joining a social network (Jain et al., 2021; Perera, 2021; Bandari, 2023;). For instance, the two factor authentication (2FA) is used by Twitter and Facebook, and some domain hosting companies, meaning that a login credential and a verification number obtained via a mobile device is used (Budiningsih et al., 2019; Jain et al., 2021; Khader et al., 2021). This process lessens the chance that an account will be compromised and restricts an attacker from stealing a genuine account and using it to the attackers' advantage.

ii. Security and Privacy Setting

Several of the social networking sites provide programmable some level of security and privacy settings, which allows users to protect their personal information from unauthorized access by third-party programs or parties (Bhatnagar & Pry, 2020; Jain et al., 2021; Perera, 2021; Quayyum et al., 2021). For example, the Facebook client allows users to change security settings and specify which individuals and group of users, such as friends, and friends of friends, is allowed to view their profile information, photos, posts, and other sensitive data (Abba & Hassan, 2022; Nawaz et al., 2022). Additionally, Facebook users have the privilege to either accept or deny third-party application access to their personal data. Internal system security measures are in place on many social networking sites. They safeguard network users against spam, fake profiles, spammers, and other threats (Abdallah et al., 2020; Perera, 2021).

iii. Reporting the User

The settings that allow users to report any abuse or rule violations by other users, thereby protecting the younger generation from harassment; for example, a user can use the report links to send a message to the person who posted something on Facebook, asking him to remove it or take it down if it offends their sensibilities but does not violate Facebook's terms (Cobbe, 2021).

2. Threat Solutions by Academic Research

i. Phishing Detection

Phishing attacks can infect a wide range of conventional web services, including blogs, social networking sites, emails, webpages, and more. Many anti-phishing techniques have been developed to detect phishing attacks. Several researchers have suggested anti-phishing practices based on techniques designed to identify phishing URLs and websites. The increasing prevalence of phishing attempts on online social networking sites has prompted the academic community to develop specialist solutions for phishing assaults in a social networking context. According to Alanezi (2021), the effective methods for identifying phishing websites are required to reduce the risks posed by phishing attacks. Because of their effectiveness in detecting phishing attacks, machine learning techniques are widely used because of the available large database and the capacity of the training models.

ii. Cyberbullying Detection

While identifying cyberbullying is more challenging than identifying spam and objectionable language, a number of scholars have tried to do so by utilising more sophisticated document representations and additional information about bullies and victims (Etuh et al., 2021). To detect cyberbullying, machine learning approaches can be used, and in order to create a model that can automatically identify instances of cyberbullying on the network, the model should make use of machine learning to identify the language patterns of bullies on social media networks. Two classifier techniques were adopted; Support Vector Machine (SVM) and Neural Network to evaluate the model, and for features extraction. The model was evaluated using several n-gram language models, and it was discovered to obtain 92.8% accuracy using a neural network with 3-grams and 90.3% accuracy using an SVM with 4-grams while combining Term Frequency-Inverse Document Frequency (TFIDF) and sentiment analysis (Etuh et al., 2021). The Neural

Network outperformed the SVM classifier, with an average f-score of 91.9% as opposed to 89.8% for SVM (Ali et al., 2019; Khan et al., 2019; Rapacz et al., 2021). In addition to words and emoticons that convey insults, obscenity, and common cyberbullying vocabulary, it can also use some other information, like the gender and personality of the participants in a suspected cyberbullying episode.

iii. Cyber Grooming Recognition

Machine learning approaches seem to be a useful strategy for combating the issue of online grooming. In order to safeguard kids from online threats, as far back as 2014, Michalopoulos et al. (2014) developed the Grooming Attack Recognition System (GARS), a method to identify, evaluate, and manage grooming attacks. By examining the child's discussions, it determines the overall risk value, which pinpoints grooming hazards to which a youngster is exposed. Risk value has a predetermined threshold, and an alarm mechanism is triggered when the total risk value exceeds it (Michalopoulos et al., 2014). This alarm system also sends a parent an immediate warning message at the same time. To alert the youngster to the level of danger in a dialogue, a coloured signal is produced.

iv. **Cyberstalking**

Devices running the latest versions of iOS and Android can be encrypted; for example, if encryption is enabled, the contents of a stolen device cannot be read by the thief (Jain et al., 2021). Furthermore, all attempts to read data from internal or external memory are blocked by the presence of a device password. Technologies that can be used to combat stalkers include firewalls, privacy guards, specialised stalker app detection software, and smartphone fingerprint lock antivirus. According to Dhillon & Smith (2019). The summary of the fundamental ways of Preventing Cyberstalking is presented:

2.5.3 Summary of Approaches to Cyberstalking Prevention

Protect Online Interaction

i. Use caution when meeting online.
ii. Reduce the use of public forums.
iii. Make sure that there are safeguards in place in online forums, e.g. end-to-end encryption.
iv. Remove any potentially harmful online chat rooms.

Increase Cyberstalking Security Procedures

i. Ensure the security of online browsing.
ii. Ensure the availability of cyberstalking prevention tools.
iii. Boost authentication measures.
iv. Check the credibility of the website.

Ensure Technical Security

i. Make an investment in safe web browsing tools.
ii. Make more use of technologies to manage login passwords and stop information theft.
iii. Monitor online security settings
iv. Develop internet filters to stop inappropriate conduct.

Develop Strong Value System

i. Maintain strong family values to avoid cyberstalking.
ii. In order to reduce cyberstalking, reduce social pressures.
iii. Increase family support for information security measures.

Define Intermediaries to Minimize Cyberstalking

i. Develop payment systems to ensure security
ii. Create pay services to protect consumer online information
iii. Increase use of personal information insurance to protect privacy
iv. Develop trust forming mechanisms to protect against cyberstalking

v. Spam Detection

In the works of Kaddoura et al. (2022), spam detection and classification requires that several processes be involved, this is as seen in Figure 2.16. The first stage ensures that data is gathered from social networking sites (Facebook, Twitter, and Instagram), as well as the use of e-mails and some online review platforms. After data collection stage, the next stage is the pre-processing phase, which includes a variety of Natural Language Processing (NLP) techniques to remove unwanted/redundant data, following this, is the third stage which involves the extraction of characteristics from the text data applying techniques such as Term Frequency-Inverse Document Frequency (TF-IDF), N-grams, and Word embedding (Khoo et al., 2020; Ali et al., 2021; Tambe et al., 2021; Thajeel et al., 2023). These feature extraction/encoding methods are meant to convert words/text into numerical vectors that may further be classified.

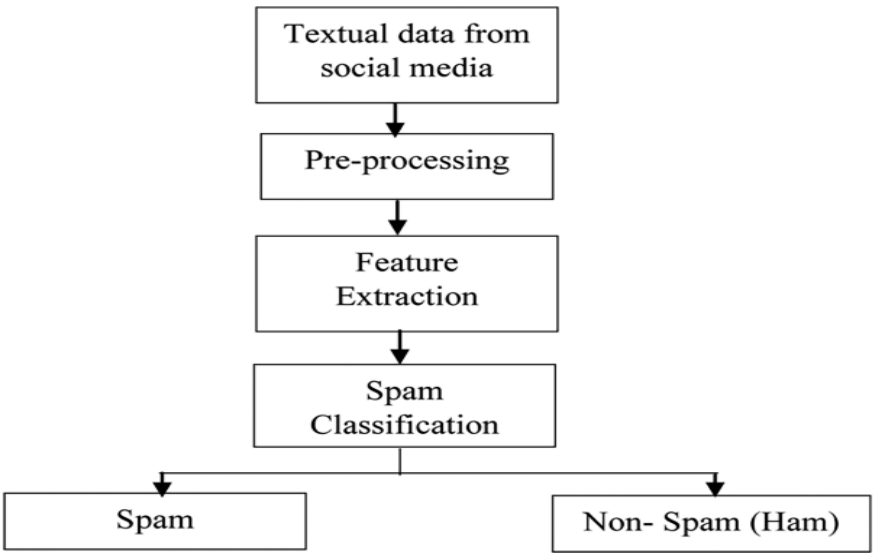


Figure 2.16: Steps involved in Spam Detection. (Kaddoura et al., 2022)

i. Malware Detection

Researchers usually create malware detection systems and usually keep record of malicious and benign software to evaluate them in sequence (Faruk et al., 2021). Faruk et al further stated that malware detection approaches can be divided into three categories: (i) Signature-based (ii) Anomaly-based and (iii) Heuristic-based malware detection. The following section will explore malware detection systems and offer the results as well as any limitations (Figure 2.17).

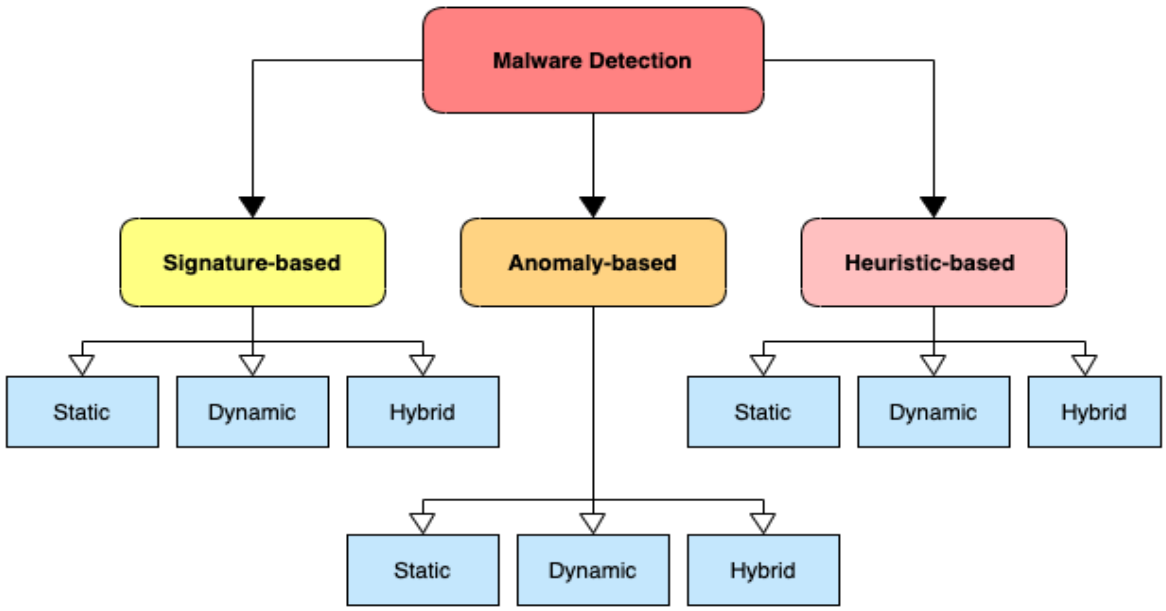


Figure 2.17: Classification of Malware Detection Techniques (Faruk et al., 2021)

From the research conducted by Faruk et al. (2021), different algorithms were deployed for the detection of malware activities by adopting innovative ideas such as Artificial Intelligence (AI), Deep Learning (DL) and Machine Learning (ML). Faruk et al emphasized on the application of AI-based approach for the detection and prevention of malware activities, and presented a comprehensive review of the prevailing state of malware detection methodologies, stating the limitations, and better ways of improving efficiency. The investigation further showed that engaging futuristic techniques for the advancement of malware detection applications would provide substantial benefits. Faruk et al are of the opinion that the conception of this combination is poised to help researchers for further research on malware detection and prevention using AI-based approach. Figure 2.18 displays the types and uses of Artificial Intelligence which consists of Machine Learning (ML) and Natural Language Processing (NLP).

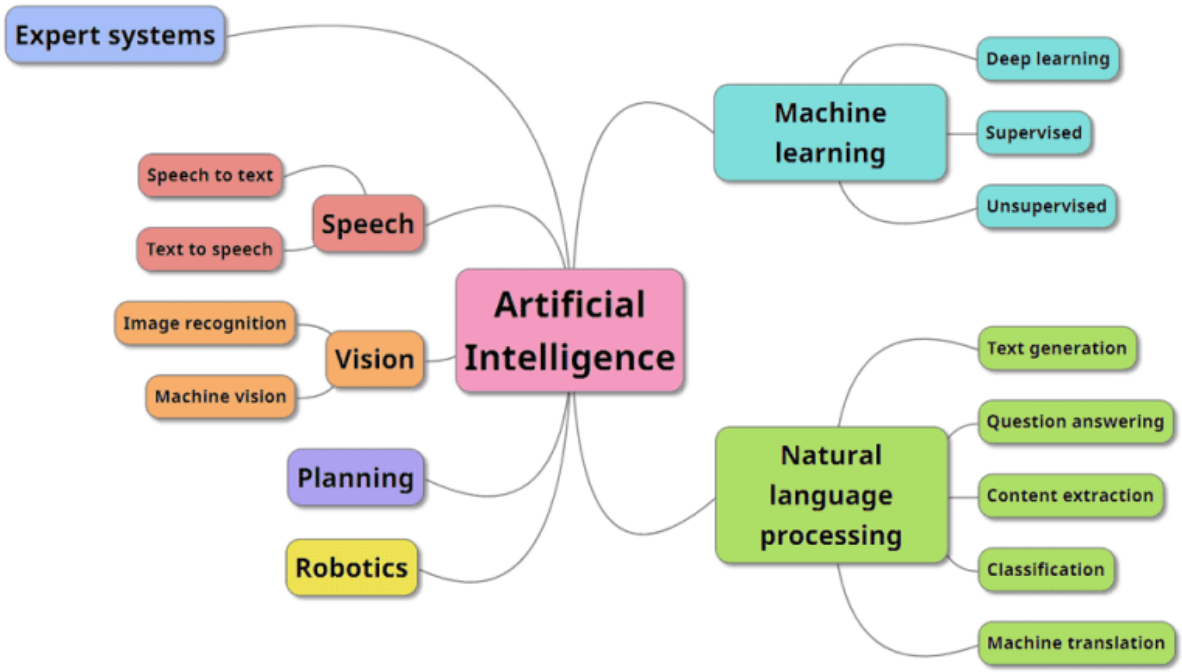


Figure 2.18: Types and Uses of Artificial Intelligence (Faruk et al., 2021)

2.6 Effectiveness of Existing Web Content Filtering Solutions

The deployment and use of content filters can be made more efficient with a greater understanding of the applications and constraints of these approaches, it is important to comprehend how the effectiveness of content filtering approaches is impacted by the setup complexity, lack of training, and non-use of system monitoring software (Turner, 2022).

According to Hounsel et al. (2019), effectively measuring of filtering systems require different components. The first to be considered is inputting the list of domains or IP addresses that are to be tested which may intensely impact results and the effectiveness of any research. The report from the Citizen Lab maintains several test lists (The citizen lab, 2019); this may include both the general lists of websites/webpages that are frequently filtered on country-based as well as globally-based lists. According to Hounsel et al. (2019), it is possible to automatically curate a culture-specific input list by examining websites that are blocked in China (Wong & Liang, 2021). They also note that the absence of a reliable block-list can make it challenging to understand the filtering system's goals, but these systems most times do not perform effectively for websites other than English Language (Cobbe, 2021).

Assessing the deployment and effectiveness of filtering technologies involve the aid of local collaborators and physical access to a sample product, manually determining a distinctive signature for a small subset of filter products, and executing network scans using the signatures to find abnormalities has been a cumbersome process (Darer, 2020). More so, monitoring the deployment constantly requires sustainable systems. This has led to only a handful of the filtered measurement systems been identified over the years. According to Ververis et al. (2021) only four filter software companies' signatures were manually constructed, and their implementation in various nations was assessed; these signatures were restricted to a certain product configuration and based on block-pages exposed by filters with public-facing IP addresses.

Khan et al. (2021) disscussed that the performance, network capacity, quality of service, and connection of the next generation of networking technologies, such as 5G and 6G will be exceptional. The combination of these technologies with big data analytics in the smart ecosystem of today will thus present huge prospects. According to the researchers, the current URL filtering methods lack fault tolerance, scalability, and real-time filtering (Ahmed et al., 2018; Adam, 2019). The researchers solved these problems by creating a machine learning-based binary classification model that is real-time, fault-tolerant, and scalable, and can handle streams of URL traffic and classify it as clean or obscene in real-time. They only employed features based on URLs, and their classification accuracy was 93% for the logistic regression classifier and 88% for the support vector machine (SVM). In just 55 seconds, the model was able to filter 2 million URLs. The precision, recall, and f1-score (the f1-score is a general measure of classification performance that takes precedence over accuracy when there is data imbalance, for

example, the number of samples belonging to one class greatly exceeds the number of samples belonging to another class) values for the suggested model were 0.92, 0.95, and 0.93 respectively. However, researchers must explore the fault-tolerant elements of URL filtering utilizing different machine learning techniques as well as the scalability challenges that may arise when the system has errors.

Adam (2019) stated that real-time fault tolerance and scalability are not supported by the URL filtering techniques currently in use. These problems are addressed in their study by creating a fault-tolerant, scalable, and real-time model to categorize streams of URL traffic. Their model's main advantage is that it uses less bandwidth, resources, and calculation time. This architecture is used in Apache Spark, which manages streaming and machine learning APIs. 2.4 million URLs total from both malicious and clean classes made up the collection. Clean URLs are identified as 1 in the training set, whereas malicious URLs are designated as 0. Resilient Distributed Datasets (RDD) from Apache Spark was used to offer distributed in-memory processing for this suggested model in a fault-tolerant way. The researchers achieved linear scalability by expanding the cluster's node count. This model scaled successfully with the Apache Spark cluster and achieved a logistic regression classifier accuracy of 96%. 2 million URLs can be filtered in 55 seconds using the Spark ML1ib logistic regression classifier. The model attained accuracy and f1-score values of 0.92, 0.95, and 0.93. The outcomes are assessed using cross-validation techniques. However, when developing spam filtering systems and real-time, scalable dangerous URLs, researchers must use URLs gleaned from streams of internet traffic analysis.

2.7 Limitations of Existing Web Content Filtering Solutions

The traditional filtering techniques such as the keyword and uniform resource locator (URL) filtering approaches have some limitations; each of these techniques only concentrates on a single kind of blocking, which does not give room for a complete view of filtering, and so because they were only tested on measurements made over a short period of time, these approaches were unable to address the challenges of collecting and analyzing continuous, and longitudinal data. More so, not a single of these techniques were intended to distinguish between localized filtering and internet service providers (ISPs) or country-wide filtering. Furthermore, keyword and URL filtering do not have apparatuses to validate filtering which may result to false

positives. In order to overcome these challenges, Raman et al. (2020) in their study, Censored Planet: An Internet-wide, Longitudinal Censorship Observatory, introduced Censored Planet, which represents worldwide and longitudinal filtering measurement platform that gathers filtering data using several remote measurement methods.

Singh (2022) noted that the existing techniques deploy common features to disallow access to content they consider inappropriate, and also consider only broad classifications of obscene materials such as nudity, sedition, among others. Nonetheless, keywords that are being sought for are hidden and could be changed without notification. Also, according to Vondersaar (2020), several of the known techniques deploy the Naïve Bayes, the classical ontology, or the bag-of-words model to identify, retrieve and categorize inappropriate content. These techniques have been noticed to be incapable of finding the needed content and expunging the unneeded. More so, Raman et al. (2020) stated that PICS (Platform for Internet Content Selection) were employed to categorize websites into whitelists and blacklists. Raman et al stated further that such systems use a universal format for labelling the data and vocabulary. Although, the Platform for Internet Content Selection-based approach is not sufficient enough at effectively classifying medically related websites due of its semantic constraints. However, to address these challenges, two main strategies are employed by Ali et al. (2019). Ali et al proposed a support vector machine (SVM) and fuzzy ontology-based adult content detection system; this method is to provide semantic knowledge for inappropriate content identification, while the SVM removes irrelevant content. The first strategy classifies the URL to provide accuracy and speed, while the second strategy has to categorize the data labels, which could represent the descriptions of the websites/webpages.

By addressing these challenges, the developed model for web content filtering will enhance the safety of social networkers, protect users from harmful and objectionable content, and contribute to a more secure and privacy-conscious internet environment, enabling individuals, employers and the online communities to navigate the internet with confidence while protecting them from inappropriate or harmful content.

2.7.1 Individual Limitations of Existing Web Content Filters

Several approaches have been proposed and developed by researchers, some of which are successful in accurately blocking problematic web content, while others are not (Ali et al., 2021;

Altarturi & Anuar, 2020). Aside the features and capabilities of the aforementioned web content filtering approaches, some limitations were identified. These are:

i. Over-blocking and under-blocking (false positives and false negatives)
ii. Easily circumvented (VPN, Proxies, language translation)
iii. The need for regular updates
iv. The challenges of override authorization
v. Technological mediocrity
vi. Implementation at ISP-level issues
vii. The need to abandon search efforts
viii. The complexities of online attacks

i. Over-blocking and Under-blocking challenges of web content filters

According to Diaz-Garcia et al. (2022), web content filtering technology is adopted to restrict users from accessing online content that violates an organisation/institution's policies. Most of the challenges that surround web content filtering technology come from the approaches that are employed to filter/block access to web content. No matter the techniques or approaches been adopted, it is generally believed that filters are not absolutely perfect solutions to the issues of web content security. Filtering technology either over-blocks and/or under-blocks web content, thereby denying access to legitimate information and/or permits access to inappropriate contents (Diaz-Garcia et al., 2022).

Considering the packet blocking approach at the ISP filtering level, the implementation of filters can be done in differing ways, such as firewalls that are to be deployed on the connections, and having the comprehensive list of websites to be blocked; this is an advantage. However, a major challenge with this filtering approach is the collateral damage that comes with it - all the known web contents on a specific IP address is blocked making it inaccessible to legitimate users, leading to over-blocking of legitimate website for legitimate users that are using the same IP address (Turner, 2021) (Figure 2.19).

In the works of Nicolaidou & Venizelou (2020), over-blocking and under-blocking rates vary depending on the websites that are utilized for testing, with some pages having a higher likelihood of errors than others. What then would be a good selection of websites to test filters on? It would be useful to know what online users try to view in order to protect themselves. In

theory, one might observe the web browsing habits of a sizable random sample of users from different age ranges to determine what websites they have access to, and then test filters on those websites. From the research so far, there are no such data, and, if it exists, they would be very expensive (Nicolaidou & Venizelou, 2020).

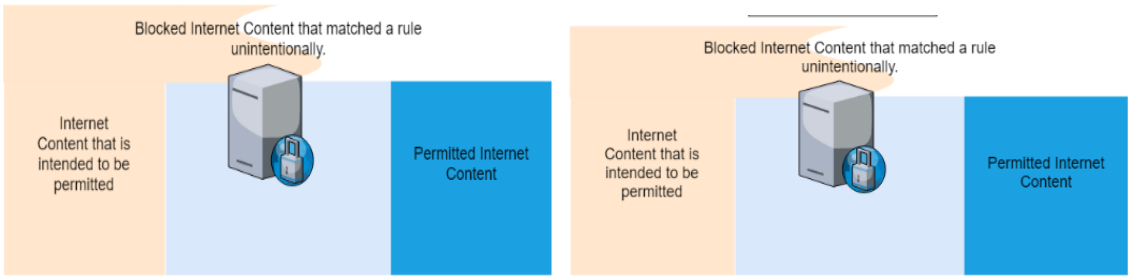


Figure 2.19: Overblocking and Underblocking Scenarios (Turner, 2021)

ii. The Challenges of Circumventing Web Filters

According to Overhaul (2022), in many nations, journalists, activists, and common users have turned to circumvention tools, such as proxy servers and virtual private networks (VPNs), which allows users to secretly or anonymously access the Internet while getting around various types of governmental censorship. In reaction, governments are progressively restricting and making the use of circumvention tools illegal, or placing restrictions on them (Overhaul, 2022). To further support this assertion, in the words of Safe (2022), there is the degree and capacity to recognize and manage tools and technologies used for circumventing any web content filtering system, for example VPN, proxy servers and DNS over HTTPS. According to the study conducted by Nash (2021) on Internet Filtering Technology and Aversive Online Experiences in Adolescents; among the analyses given is that some social networkers are more technically inclined than others, and so the authors foresaw that effectiveness of filtering technology curbing aversive online experiences would be lessened for such users who are tech savvy are able to circumvent the filtering technology.

Other techniques, such as using different protocols (such as FTP, telnet, or HTTPS) or carrying out searches in a different language, allow users to get around filters (Turner, 2021). By translating a restricted Web page into a language that the filter does not support, language translation can be used to trick the filter (Chhibbar, 2022). Even if users are unable to use external proxies, they can still bypass restrictions by using simple techniques such as using

search engines like Google to obtain cached versions of restricted Web pages (Aktay, 2018). Multilingual compatibility and a wide range of Internet protocols are features of efficient filtering software (Turner, 2021). The additional issue for filtering solutions is to examine email traffic and decide whether to prohibit or allow it depending on the content filtering policy.

iii. The Need for Regular Manual Updates of Web Page Lists

In the words of Ali et al. (2021), Uniform Resource Locator (URL) based filtering approach disapproves or approves access to content by way of comparing between the web page's domain (and IP address equivalent) that was requested and the domains in a list that has been kept. This method, however, can only identify the websites on the list and calls for the implementation of a URL list. Additionally, the system's accuracy would rapidly decline if the list is not routinely updated due to the quick proliferation of new websites. Most domain blocking web filtering solutions actively search for problematic websites that could be added to the blacklist using human review teams. Then, this updates the local copy of the list, and this list is made available for download; once more, this takes a significant amount of time and resources (Ali et al., 2021). It is important for planned checks to be more in order to ensure these updates are happening on a regular basis. This is to avoid obsolete information being accessed.

iv. Override Authorization

Most of the known content filtering approaches have the option settings that may permit only authorised personnel to bypass the settings of the web content filter; this approach is prevalent in cases where the computers are supervised, and where it was observed that the filters are overly performing the blocking. In most case, the company executives and owners have this rare privilege to act in that manner.

v. Technological Mediocrity: In the research conducted by Maserumule (2020), majority of the parents interviewed on web content filtering showed that they were ignorant of any content filtering software and, therefore, do not use it to protect or monitor their children's online activities, but claimed the use of other strategies, such as words of mouth, and some online safety tips, especially on cyber security. The study also reviewed that parents would appreciate any support in form of training, creating awareness, and information about objectionable websites, with regards to the use and clear purpose of content filtering software and techniques to monitor their ward's online activities. However, the exploratory analyses conducted by Nash (2021)

revealed that ignorance about web content filtering use was not associated with kids' age, gender, or did location affect the level of knowledge. More so, online users who do not understand the risks and the impact of the online risks think it is nothing to worry about.

vi. ISP level Web Filtering Implementation Issues

When it comes to the implementation of web filtering at the ISP-level based filtering approach, there are several issues that needs attention (On et al., 2019). These are: a named and known website can be renamed easily making the site names not been able to match with the list on the database (white or black lists). Language translation: Some websites do not have English language content, so filtering such websites may produces errors; the resultant effect of this is that international websites may not be effectively filtered (Steingr et al., 2023). For the database to be highly effective, such dataset must contain domain names and IP addresses. Unfortunately, not all Internet users have access to the Internet via an ISP; some use broadband connectivity and other means to connect to the Internet. So many websites have mirrors and different URLs and if these are not included in the blacklists, then the filtering process can be bypassed.

vii. The Need to Abandon Search Efforts

According to the study conducted by Jamali (2018) on the effects of Internet filtering on users' information-seeking behaviour and emotions, the author stated that some participants could easily give up their searching effort if faced with a stringent filtering system, particularly if the sought information was not of much importance. Jamali further explained that the idea of using devising different approaches to handle filtering was different among participants; for instance, while some tried a different search strategy before using anti-filter, some others are of the opinion that if anti-filter software was not effective, then their search strategy would have to be changed.

viii. The Complexity of Online Attacks

The complexity of attacks grow in tandem with the complexity of the filtering algorithm, and adversarial attacks have been used to test automated filters (Giovanni Sartor & Andrea Loreggia, 2020). This method employs two interacting systems, a discriminator and a generator, both of which are typically implemented using a neural network. By delivering communications that the discriminator misclassifies, spam messages that are mistakenly classed as non-spam or phoney

news or reviews that are categorised as original, the generator seeks to fool the discriminator (this is the goal of the generator). The filtering system in question is then attacked using messages that are able to trick the discriminator. Both automated systems and humans find it very challenging to identify fraudulent content produced by adversarial networks.

2.8 Web Browsers

As an online user, a web browser is a piece of software or an important application program majorly used for accessing and performing different activities on the Internet, like browsing the Internet, sending and receiving emails, visiting social media platforms, buying and selling of products and services, uploading and downloading of files of different formats, and so cybercriminals use web browsers to commit Internet crimes (Mugisha, 2019). Figure 2.20 shows that so many browsers exist such as Google Chrome, Mozilla Firefox, Microsoft Edge or Internet Explorer, Opera, and Apple Safari; of all these browsers, Google Chrome seems to be the most popular among the Internet user community (Jillepalli et al., 2017). Jain (2018), further explained that a web browser is a piece of computer software used for locating and displaying web sites. Web browsers are the interface between and online user and the web server, hence, web users can access data from web servers.

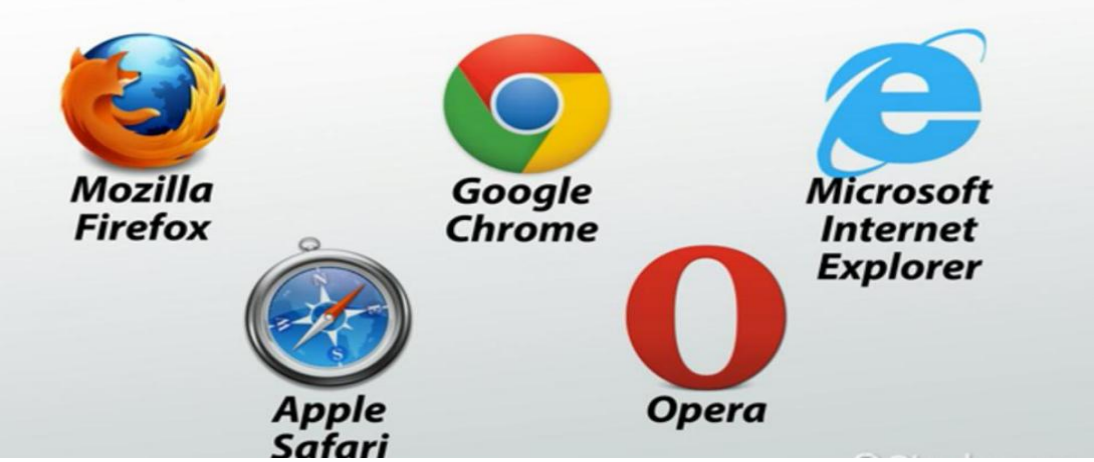


Figure 2.20: Types of Web Browser

2.8.1 Functionalities of a Web Browser

Browsers are divided into three primary sections: the interpreter, the client software, and the controller (Figure 2.21). The other two components; the client program and the interpreter are managed by the controller.

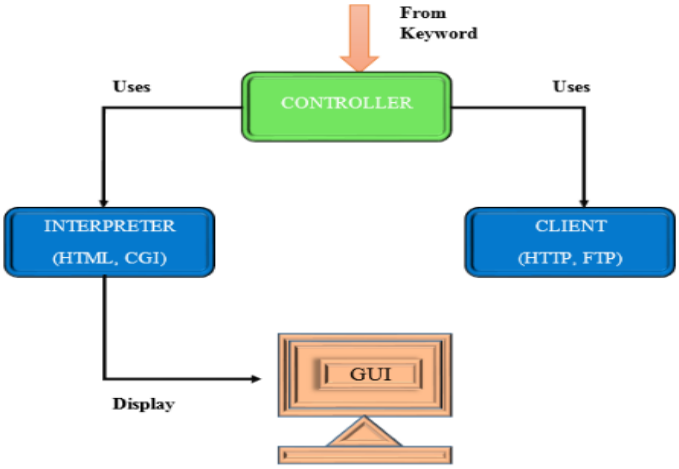


Figure 2.21: Functionalities of a web browser (Jain, 2018)

A controller accesses any document via a client program (HTTP, and FTP) after receiving inputs from conventional input devices of the user. The controller uses an interpreter (HTML, JAVA, CGI, etc.) to show the page on the screen as soon as it is accessed. As such, it serves as a user interface for the World Wide Web. Rendering content in markup languages to show web pages is the main use case for web browsers, and the Hyper Text Markup Language (HTML), an international W3C standard is the markup language used when discussing the Internet.

2.8.2 Basic Architecture of Web Browsers

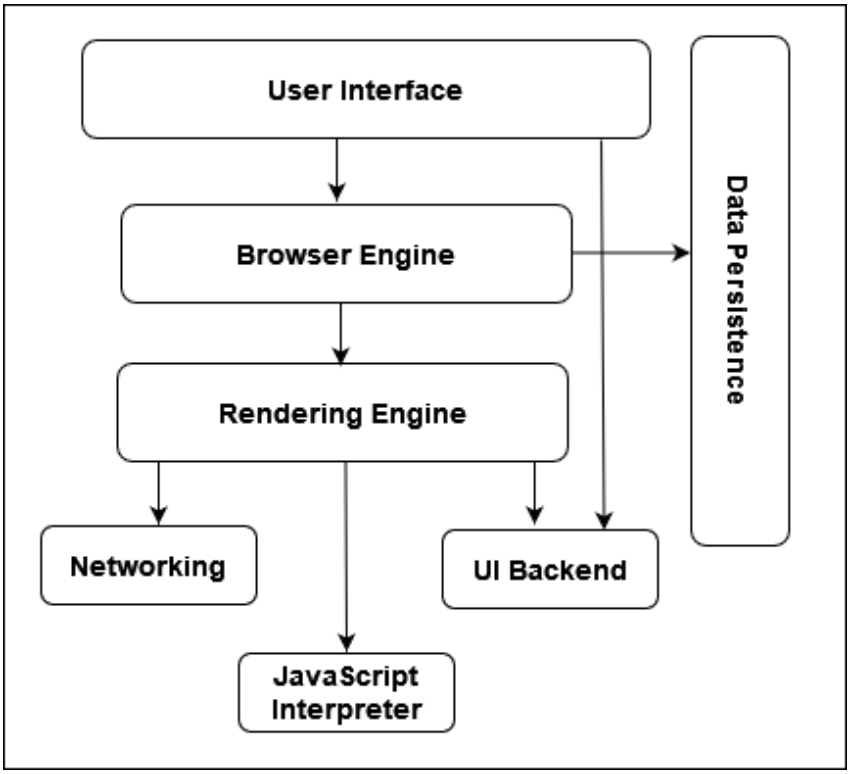


Figure 2.22: Basic Architecture of Web Browser (Jain, 2018)

Figures 2.22 and 2.23 shows the basic architectures of a web browser and specifically, the Google chrome browser.

i. The User Interface

This is the space where the interaction between a user and the browser takes place. Some of these spaces include the address bar, the next and the back button, the home button, refresh and stop, bookmarking options of webpages.

ii. The Browser Engine

This serves as a link between the rendering engine and the user interface. The browser engine's primary job is to query and control the rendering engine using input from different user interfaces (Bui, 2016.)

iii. Rendering Engine

The rendering engine is responsible for rendering/parsing the requested webpage on the user's browser screen. It interprets HTML, XML documents, including images that were formatted

using CSS and further generates the layout which should be displayed on the user interface. It is important to note that different browsers have different rendering engines. The key components that make up this phase are the HTML and the CSS parse.

iv. Networking

The networking component of any web browser is responsible for all the different aspects of security and internal communication. It retrieves the URLs using Internet protocols such as the HTTP or the FTP, for instance, the network component sends the HTTP request to the server to be displayed on the user's browser. It also reduces traffic by implementing a cache of retrieved documents.

v. JavaScript Interpreter

This is the component of the web browser that is written to interpret and executes the JavaScript code embedded in a webpage, which is later sent to the rendering engine for onward display.

vi. UI Backend

This component is majorly for drawing essential widgets such as the combo boxes, and the windows and it reveals a general interface. The UI backend uses operating system user interface methods.

vii. Data Persistence

Data persistence, otherwise known as the storage is the small database created on the local drive of the user's computer where the browser is installed; user data such as cache, bookmarks, cookies, preferences, etc., are managed.

2.8.3 Architecture of the Google Chrome Web Browser

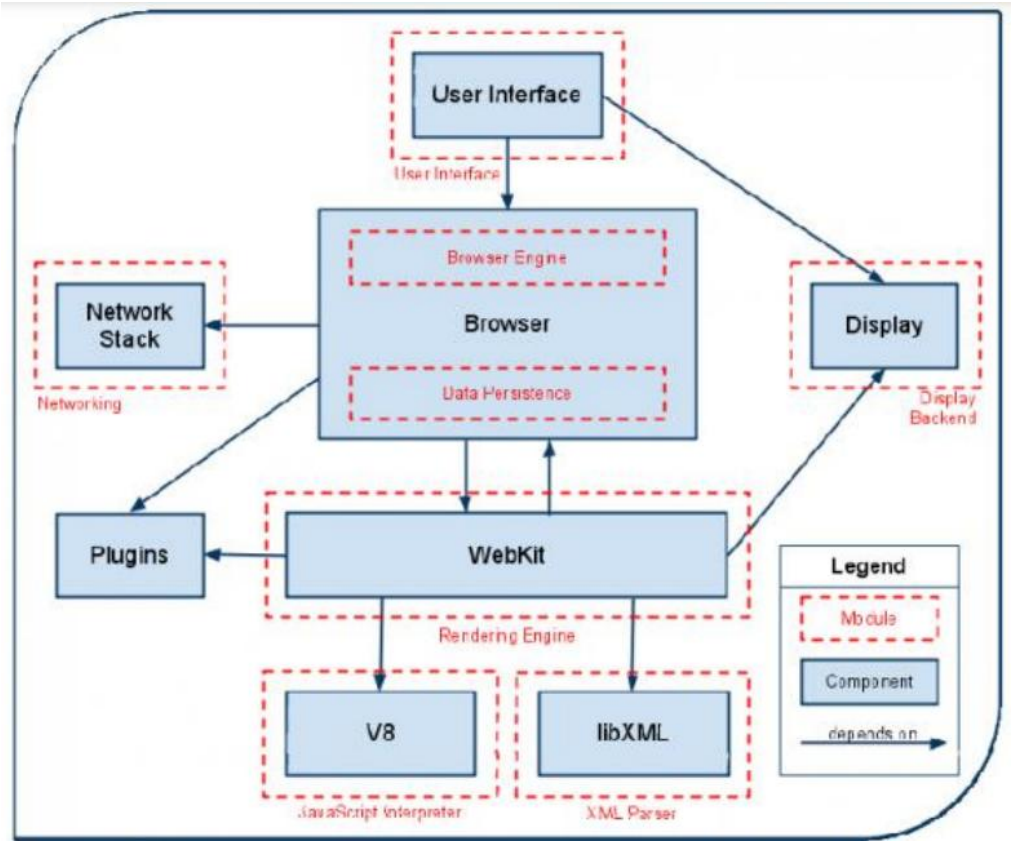


Figure 2:23: Architecture of Google Chrome Web Browser

2.8.4 Web Browser Extension

Small JavaScript programs known as browser extensions, also called plugins or add-ons are used to give web browsers more features and capabilities (Gnana & Kamalanaban, 2016). For their improved user interface and practical features, extensions are a crucial component of web browsers. JavaScript, CSS, and HTML can all be used to develop extensions. They can use cross-origin XMLHTTPRequests or content scripts to communicate with servers or webpages. The majority of extensions are designed to give web browsers a specific function, such as web search, download management, or email notifications. In this study, the purpose of web browser extensions is to protect users' online behaviour.

Browser extensions, like the Egenti-Filter extends the functionality of a browser, usually in the form of context menus, extra toolbars, productivity tools or customisations to the browser's user interface. There are different formats of extensions, such as executable plugins or add-ons to perform or execute specific actions, such as PDF readers, and Flash players. Extensions also

presents different set of functionalities, for example, displaying specific information based on user preference, customising the active webpage, accessing and manipulating security and the privacy of sensitive data, developing and debugging web applications. These are the functions (section 4.6) performed by Egenti-Filter browser extension that was developed to achieve the aim of this research endeavour.

2.8.5 The Role of Extensions in Web Browser

Browser extensions also referred to as plug-ins or add-ons are small pieces of code that allows programmers to enhance the functionality of any browser. By adding functionality, browser extensions enable browser customisation. There are significant differences on how these extensions are implemented across the different browsers. By altering the browser's user interface and how it interacts with webpages, browser extensions change the fundamental browser user experience (Hausknecht et al., 2015).

Browser extensions enable for browser customisation by adding functionality, which is set to perform specific actions on browsers. The manner in which these extensions are integrated varies greatly amongst the four major browsers (Jain, 2018; Mugisha, 2019), the ability to add themes to the browser is one example of this variety. This feature is available in Firefox and Chrome, but not in Internet Explorer or Safari; although, this example clearly demonstrates the differences in extensibility between the four browsers. Browser extensions alter the primary browser user experience by altering the user interface and interacting with websites.

For this research, the ‘browser extension’ was encapsulated within a manifest.json file, which explicitly conveys the extension's structure, this is so because it is required for any web browser extension, and it contains important information that defines the extension and offers the browser the necessary information to load and run the browser extension. The manifest.json file also allows developers to declare assets, such as images and scripts that are used by the extension. It also includes an index.html file that serves as the user interface component, designed to display pertinent information when a blacklisted site is accessed by the user.

2.8.6 Benefits of working with JSON Files

Structured JSON files offer several benefits (Khder, 2021), as listed:

Simplicity and readability: JSON employs a human-readable format of key-value pairs and arrays, making it simple to write and understand. Unlike other data formats such as XML, it does not require any particular tags, attributes, or schemas.

Compact and efficient format: JSON syntax allows for simple data parsing and even faster implementation. It frequently requires less data to download than other formats, and it can be understood in most modern languages without the use of additional libraries.

Compatibility and interoperability: JSON is extensively supported by most modern browsers, web servers, and online APIs, making it simple to transmit data across diverse systems and contexts. It can also be used in conjunction with other libraries and tools that provide functions for parsing, validating, manipulating, and modifying JSON data (Bhavaraju et al., 2018).

Performance and efficiency: JSON is often faster and lighter than other data formats, such as XML, due to its reduced size and simpler structure. It also has a simpler and more consistent syntax, making it easier to interpret and edit.

Self-describing: JSON is self-describing which means that it provides metadata that describes the data it contains. This makes it simpler to comprehend and operate with.

Adaptable structure: JSON files have an adaptable structure since they store data in nested objects and arrays that include values. This structure is very customizable, and the columns inside the data source do not prevent additional data from being added to the collection.

The manner in which these extensions are integrated varies greatly amongst the four major browsers, namely Google Chrome, Mozilla Firefox, Apple Safari, and Internet Explorer. It is a little bit of code that allows developers to extend the capabilities of the browser, and it is also known as a plug-in or an add-on (Kareem Thajeel Thajeel et al., 2023; Mathews & Chimalakonda, 2021). The plugin enables users to change the browser behaviour on specific websites. To retain the idea of least privileges in the browser, a set of rules must have to be developed (Jain, 2018), and while extension usage is common on the desktop, it is still in its infancy on mobile devices (Bhavaraju et al., 2018).

2.8.6 Browser Extension Cyber Threats

There are three major likely attacks on web browser extensions:

i. Malicious Extensions: A cybercriminal could install a malicious extension in the browser, and this could result in massive damage. According to Jain (2018), a few malicious extensions have been reportedly found in the wild before, which were used to steal passwords for banking and other High-Security websites. However, these attacks were immediately blacklisted by the web browsers, but the threat is still there. There are instances where some malicious extension installation mechanisms can be silently installed. To address this challenge, both Google Chrome and Mozilla Firefox browsers provide a method to install extensions silently. However, these browsers display a confirmation dialog to the user before the installation is complete.

ii. Extension Vulnerabilities: The web browser extension on its own may suffer from some form of vulnerabilities; this could be due to insecure coding practices. The study conducted by Carlini et al. (2012), reported that even the most popular extensions, plus the ones developed by Google, are vulnerable to attacks, which could be exploited by malicious websites. Among this list are improper handling of user input, poor design, unstructured input being used in the extension, insecure configuration, metadata attacks on the extension, and poor implementation.

iii. Non-Malicious Extension

a. Network Attackers: Online users who use insecure networks are prone to network attackers. Network attackers aim to steal personally identifiable information (PII) or credentials from a target user; for this to be accomplished, the network attacker will have to read and alter HTTP traffic to enable the launching of a man-in-middle attack.

b. Web Attackers: An online user may visit websites which has ads (advertisements), and in doing so, the website triggers the cross-site scripting (CSS) attack on an extension if the extension identifies the website's data or functions as being trusted. The main aim of a web attacker is to gain access to the browser user data, such as browsing history, or user's credentials.

2.9 Review of Related Works

Several Web applications now offer web content filtering as a service for parental control and regulating Web content access for users connected to enterprise, library, and school networks; (Deotte et al., 2021; Gómez et al., 2019; Chrysomalidis et al., 2021), SquidGuard (Ali et al., 2021; Chrysomalidis et al., 2021), and DansGuardian (Gómez et al., 2019; Nicoletti, 2013; Turner, 2021), are some of the most popular Internet filters. All of these applications use a list-based approach, and some (Dans-Guardian) also support content analysis and/or Platform for Internet Content Selection **(**PICS-based filtering) (Ali et al., 2019). They are, however, not based on a Web content filtering model; instead, they integrate and optimize recognized methods in firewall blocking and data mining and other fields to provide a service that addresses the needs of specific user categories while maintaining a high level of efficiency. While this is one of the most important features that makes such applications usable, it has the significant disadvantage of requiring quite restrictive filtering. None of them support more sophisticated knowledge representation tools than the available platform for internet content selection (PICS) vocabularies. Furthermore, user characteristics are not taken into consideration in policy specification because the same policies apply to all users. Web content filtering, on the other hand, is similar to access control approaches in that policies are based on the characteristics of the subjects and/or objects. Web content filtering can be thought of as an extension of access control because it can be used to protect both objects and subjects from unauthorized access (Kashmar et al., 2021).

According to Raman et al. (2020) there has been plethora of literature that has attracted a continuous research interest on web content filtering across the globe. According to Vondersaar (2020), Web content filtering has continued to be a fiercely discussed subject of interest because of the strict dependence on technology, especially the Internet for teaching and learning, and so, social networking continues to grow exponentially (Drăghici, 2022). Caught directly in the midst of this discussion are teachers, students, and social networkers who use the Internet daily.

Many countries have attempted to limit the open access of the Internet through Internet filtering. As a result, numerous studies have been done to investigate filtering methods and mechanisms in various countries worldwide. In this part, we survey a few of these researches and explain their relevance to this work where applicable. Researchers, who reside in nations where internet

filtering is functional, may be concerned about the repercussions of conducting research on internet filtering; this could be one explanation for the paucity of studies on web filtering from an information standpoint.

A substantial body of literature has recognized the popularity of web content filtering (Altarturi & Anuar, 2020; Baishya & Kakoty, 2019b; Cobbe, 2021; Jain et al., 2021; Mathews & Chimalakonda, 2021; Turner, 2022; Duncan & Chen, 2023). Researchers have therefore long reported internet filtering in different countries including Afghanistan (Acharya et al., 2019), Russia (Ramesh et al., 2020), and Spain (Ververis et al., 2021). However, these studies put together are far less compared to those conducted for the Great Firewall of China (Beznazwy, 2019; Hoang et al., 2021; Khandkar & Hanawal, 2021). Although, the Domain Name System (DNS) filtering device used for the China's Great Firewall (GFW) has progressed significantly over the past years, however, several previous researches focusing on China's DNS filtering techniques were implemented over short time periods, which led to unobserved variations in the GFW's behaviour (Hoang et al., 2021).

i. Related work on Country-Specific Internet Filtering

The fundamental purpose for filtering is thought to be political in several countries. For example, a publicly available dataset of websites was used to assess accessibility in Pakistan, it was discovered that the government filters at the DNS and HTTP levels, and that the widely used Tor network which allows online users to surf the web anonymously was mostly accessible (Internet Censorship in Pakistan: Findings from 2014-2017, 2020). Evidence abounds that in Iran, network traffic is being directed to centralized equipment which is under the full control of the government. As at July 2022, two months before the protests started, Iranian internet users' search results have been filtered and limited by Google web search engines since they are classified as underage users. 85 million Iranians are now regarded as children after a protracted period of network disruption and a drop in internet quality. It was also gathered that the filtering was not implemented by the operators but by the Ministry of Communications.

The work of Thangaraj (2019) focused on an algorithm for filtering web content that was referred to as An Early Decision Algorithm. This algorithm quickens the filtering process in the web content, and a decision is made either to block or allow the web pages. It was implemented using the testing samples of DansGuardian. The focus of this work was to address the challenge

of delay period from textual categorization algorithms to perform the runtime content analysis of web content. Although, this aim was achieved, however, it did not combine with more keywords and also, there was the need to maintain the URL list; this again, is a major concern. Again, the study by Gupta & Hilal (2019) takes a different perspective, as it gives an interface study which is of immense benefit to parents, employers, teachers, online users, and children to surf the Internet in more secure ways and make pleasurable online interaction. The import of this algorithm is to ensure that the online users, especially the children, focus on the website by not performing any blocking or filtering process, rather by redirecting their interest to educational aspects of the site.

Majoring into another dimension of content filtering such as the filtering of fake news on the Internet, Kumari et al. (2021) stated that information or tales that have been purposefully manufactured to deceive or mislead the reader is known as fake news or disinformation. These days, social media platforms have developed into the perfect environment for the fast spreading of false information, causing confusion, fear, and possible health risks among users. The Natural Language Processing (NLP) field faces the most urgent issues as a result of the widespread and quick propagation of false information. Therefore, it seems like two crucial tasks to tightly combine with disinformation detection are assessing the novelty of the news item and assessing the reader's emotional state after reading it. To the best of the researchers' knowledge, previous work did not investigate misinformation detection with mutual learning for novelty detection and emotion recognition. Their study has the notion that coupled learning of novelty and emotion from the target text is a powerful tool for spotting false information. The researchers suggest a deep multitask learning framework that can simultaneously recognize emotions, identify novelty, and identify false information.

In furtherance to the assertions by Kumari et al. (2021), Ozbay & Alatas (2020) opined that the way news is created and disseminated was different as a result of technology use, especially the Internet and the creation and widespread use of social networking sites which has made news more timely, affordable, and accessible. However, there are some drawbacks to this shift; particularly risky content or obscene material such as user-generated false news on social media. This research by Ozbay & Alatas (2020) focuses on false news and proposes a two-step strategy for recognizing it on social media. The method's original phase has to do with pre-processing of the dataset to turn unstructured datasets into a structured one. Using the discovered term

frequency (TF) weighting technique and document-term matrix, the textual content in the dataset containing the news are being represented as vectors.

A similar to this our study was presented by Johnson et al. (2020); the article presents the same challenge as in this thesis; a binary classifier detecting if the URL is malicious or benign; the study compared the performance of traditional machine learning algorithms, such as Random Forest, Classification and Regression Trees (CART), and k-Nearest Neighbors (kNN) against popular deep learning framework models, such as Fast.ai and Keras-TensorFlow across CPU, GPU, and TPU architectures. However, it added a second multi-class classification that detects the type of attack, but obtains a far more complicated neural network and obtains almost the same accuracy with ours.

According to the research conducted by Mathews & Chimalakonda (2021) on 'Detox Browser - Towards Filtering Sensitive Content On the Web', the researchers aimed at improving the mental health of users by offering the ability to take control of the content available to viewers on the Internet; however, their experiment did not perform as expected, as it caused lots of false positive results. To resolve this challenge, the researchers tried to analyse the whole content, however, this approach tends to be more resource-intensive, and so the approach was abandoned. Another major setback was that the categorizer was trained on Indian headlines; as a result, the experiment is region-specific. Their experiment also works only in the English language and the tool does not support local languages. However, it was believed that the tool used could be adopted by the public, but the researchers could not comment on the efficacy of the experiment. Another limitation of this approach is that there is a need to specify patterns for the elements to be analyzed and replaced appropriately, meaning that the addition of new websites requires manual effort, and also major updates to how the content is displayed on the target website can break the primary scripts; for this to be resolved, the researchers have to keep checks on websites that display warning messages when the patterns no longer return the expected results.

In the works of Norma Guti´errez (2021), the authors cited Chiba et al. (2012), and stated that Chiba et al proposed a system that could detect malicious or benign websites by only analyzing the IP characteristics; the researchers created a dataset extracting only campus traffic, and preprocessed the data by separating the IP addressed into bits and applied two different Machine Learning algorithms; the Support Vector Machine (SVM) and Restricted Boltzmann Machines

(RBM). It was gathered that they achieve a maximum of 90% accuracy. Although, we used the support vector machine, but we did not use IP addresses, rather we used URLs which gave more features and produced a more precise model than that proposed by Chiba et al. (2012).

Furthermore, in the works of Xuan & Nguyen, (2020), the researchers used the URL features to extract dynamic behaviours of the URLs which were trained with two supervised Machine Learning algorithms; Support Vector Machine (SVM) and Random Forest (RF). The contrast between this and this study is that far more URL features were used than we did, which also increased the computational resources of their network. Interestingly, they equally obtained high performance accuracy rate of 93%, having a 3% less Recall rate from ours (96%).

2.10 Summary/Meta-Analysis of Reviewed Works

The literature review suggests that several of the filtering systems are based on exclusion filtering approaches; it shows high error rates in most of the existing filters. Exclusion filtering is becoming frequent and it is based on black-lists of recognised websites and/or webpages. Exclusion filtering adheres to the tenet 'innocent until proven guilty' and permits all content to pass through. The benefits of web content filtering cuts across different stakeholders, such as parents, learners, and employers. Different deployments platforms exist for the implementation of web content filters, such as ISP, third party application, the Government, school libraries and individuals. Unclassified websites and content may escape the filtering system, and bearing in mind, the enormity of content on the Internet, several undesirable content may pass through the filtering technology unnoticed. These filtering systems depend so much on having a comprehensive updated database list, thereby, resulting to high rates of false positives and negatives. There is the challenge of website translation and content classification which should improve the quality of search results. Only textual contents are filtered leaving images and photos. Some experiments were not carried out only customers' reviews were used for analysis. The existing systems are easily circumvented, and there is the challenge of different languages and contexts that are not in English language.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Introduction

This study adopted the developmental research design which focuses on the design, development, and evaluation of an improved web browser extension filtering model (Egenti-Filter). The research combines both qualitative and quantitative approaches, where quantitative data was collected through the distribution of a questionnaire to answer the research questions, and qualitative data was gathered through user feedback and testing of the model.

3.2 Research Methods used in this Study

The methodology applied in this research work is the Prototyping Model; which is a development methodology popularly used in design, research, and software development to explore concepts/ideas, solve problems, and enhance solutions by crafting simplified versions of the end product (Bjarnason et al., 2023). These final versions are referred to as prototypes, which allow team members to test, evaluate, and iterate on the proposed designs or ideas/concepts before commencing the full-scale development of such a project. In essence, the prototyping model displays the functionality of the proposed project that is under development, however, it does not hold the actual logic of the original software. It enables the easy understanding of user requirements at an early and initial development stage. Getting valuable feedbacks from users and help both researchers and developers have a better understanding on the exact output of the final product; hence trying out before implementation.

Early issue identification makes the prototyping model to save time and resources. There are different stages of prototyping model (Hu et al., 2024;Wulandari et al., 2021);

i. The basic requirement identification stage which involves understanding the basic requirements that has to do with user interface, other aspects such as performance and security may be ignored at this initial stage.

ii. The next stage is the development of the initial Prototype; where the basic requirements are presented and user interfaces are delivered. At the stage, most of the features may exactly now work as expected, however, it gives a similar look and feel as is expected in the final product.

iii. The third stage is the Review of the Prototype, here, the developed prototype is presented to the client and the other stakeholders involved in the project. At this stage, the feedback is collected for further improvements of the product.

iv. The final stage is to Revise and enhance the Prototype; at this stage, the feedback and the review comments received from the previous stage are discussed and some important factors (time, budget constraints, technical feasibility of the actual implementation. The agreed upon changes are accepted are yet again integrated in the new developed Prototype, and this cycle continues until the client's expectations are met.

The practice of adopting the prototyping model with the human-computer interaction domain is well established and it is used to explore and test new concepts or ideas and solutions for user interface designs. Similarly, with startups, using prototype model ensures that an early version of the final product is usually used to validate product ideas with clients; for instance, by presenting mockups to clients, which is cost effective in obtaining feedback (Alves et al., 2020).

3.3 Development of Web Browser Extension Filtering Model

This model was developed to serve as an effective real-time web content filter towards improving Internet users' online experience by blocking access to malicious, fictitious or objectionable websites, thereby preventing cyber-attacks (identity theft, malware infections, phishing schemes, etc.,). The model was developed using the following combination of machine learning algorithms and predefined rules to detect and block malicious content:

- Step 1: Requirements Gathering: Initial research was conducted to understand common security threats faced by users and the limitations of current web content filters.
- Step 2: Model Design**:** The filtering model was designed to scrutinize URLs for signs of malicious intents (phishing, malware, etc.).
- Step 3: Implementation**:** The browser extension was implemented using JavaScript and integrated with the browser's API to filter content in real-time.

- Step 4: Testing and Refinement**:** The extension was tested with a variety of websites to assess its accuracy in identifying malicious content. Based on testing feedback, the model was iteratively improved.

These following sections present the components involved in the design and development of the browser extension (Egenti-Filter). Figure 3.1 reveals the modules that are involved in each of the research phases. This stage is in six phases, with each stage employing a different approach. The phases are mutually dependent as each is built on the outcomes of the previous stage.

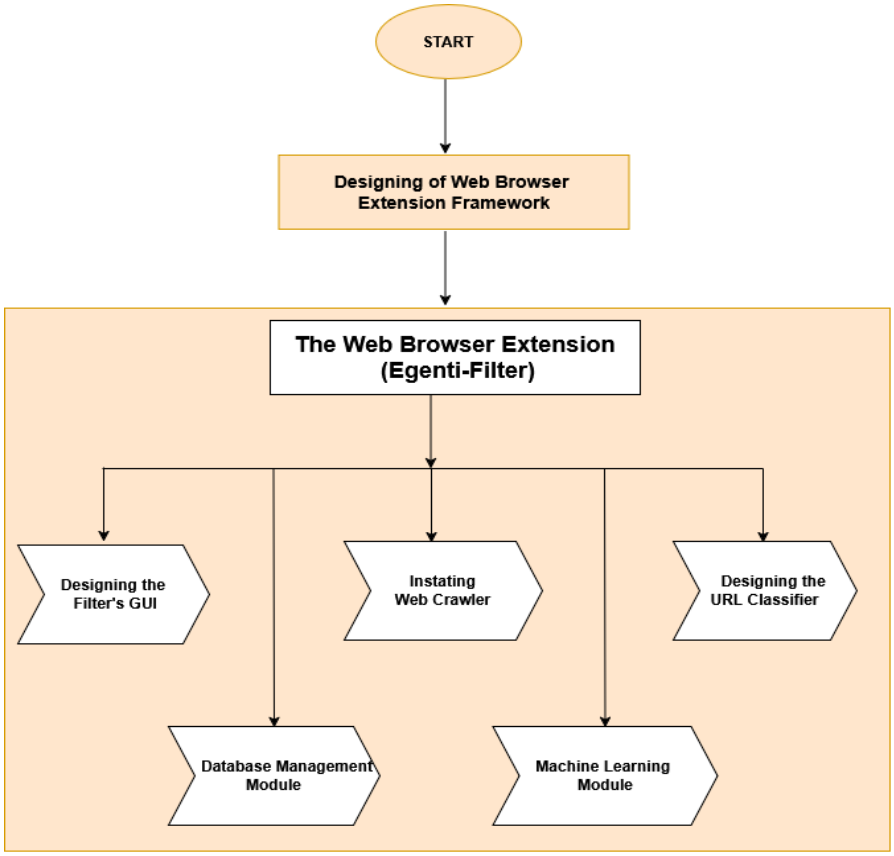


Figure 3.1: The phases of the design

The model comprises of different components such the web crawler, the database management system, the URL classification and the Graphical user interface (GUI) - for updating of the database. This section focuses on six phases;

i. Inventing Web Crawler for Link Extraction
ii. Designing URL Classifier
iii. Machine Learning Module
iv. Database Management Module

v. Designing the Filter's Graphical User Interface (GUI)

vi. Framework for Web Browser Extension (Egenti-Filter)

3.3.1 Inventing a Web Crawler for Link Extraction

A Web crawler is an effective tool for gathering information from the web by locating every URL for a single or several domains (Khder, 2021). A web crawler begins with a seed, or list of websites to visit. The crawler locates links in the HTML for each URL, applies certain filters to those links, and adds the newly identified links to a queue. For this project, when any link is parsed to the Crawler Application, it opens the page, downloads the content of the page, and starts looking for all links on the page, the links are fetched one after the other, and are returned to the Database, named CrawledLinks file. In essence, the Python Regular Expression (RegEx) operation looks out for all the expressions that qualify as a link, and returns the URLs. For this project, the web crawler application is designed to crawl webpages as well as documents. The minimum number of times for crawling any website is 5seconds. The Web crawler flow is depicted in the Figure 3.2.

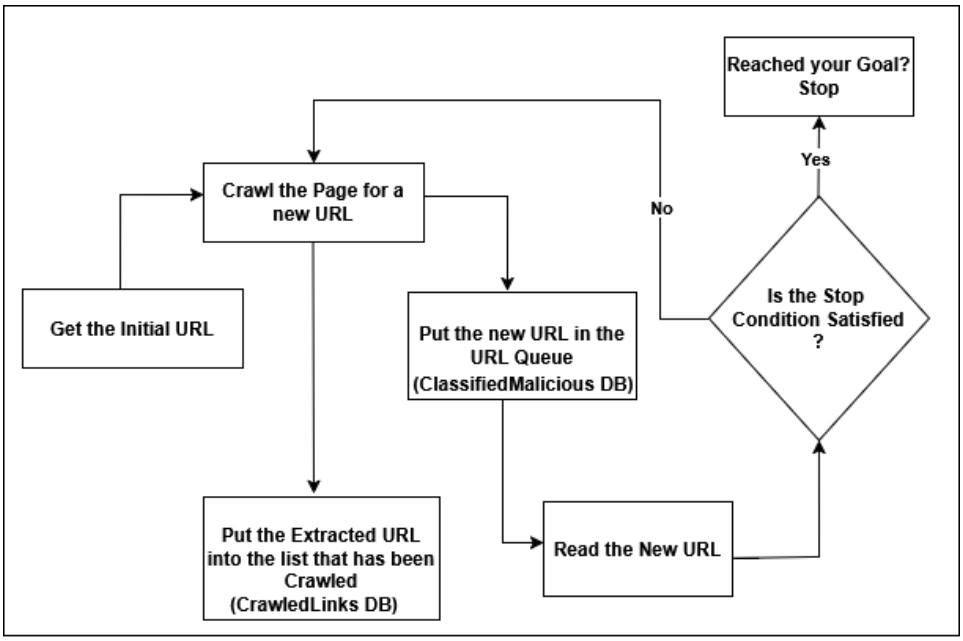


Figure 3.2 Web Crawler Workflow

After the website has been crawled, the application presents a question to the administrator, if the crawled links are to be classified as malign or benign, the user is also prompted to either crawl for more links or to terminate the crawling process. Whatever is the case, upon hitting any key on the keyboard, the crawler application terminates, and the URL classifier is triggered to

commence its operation. Thereafter, the number of malicious links is presented to the administrator. Upon termination of the process, the GUI is presented with Crawled and classified links for further processing depending on what the administrator wants to achieve; for example, updating of the backlisted database. The Crawler takes it augment from the argument file to know if it is to crawl a website or a document. The malicious links are temporary kept in a database file known as ClassifiedMalicious. Also, when the crawler application crawls the Internet for links, the links that are fetched are temporary saved in the CrawledLinks file.

3.3.1.1 Technology used in developing the Web Crawler Application

Scikit-learn (sklearn) Libraries: With its foundation in Scipy, Numpy, and matplotlib, the Python module Scikit-learn is a useful tool for predictive data analysis (Hao & Ho, 2019). The public can utilize this state-of-the-art software for free. Scikit-learn make it simple to handle common machine learning problems like clustering, regression, and classification. Scikit-learn perform better for some machine learning classifiers than other Python machine learning libraries, such as Support Vector Classification, Random Forest Classifier and Decision Tree models are the ones that were employed from the Scikit-learn module.

Two standard libraries are used for the development of the web crawler: The Requests and BeautifulSoup. The Requests and Beautifulsoup4 provide a straightforward way to connect to the Internet. (A Python module called Beautiful Soup is used to extract data from HTML and other markup languages. Beautiful Soup advises the user to extract specified content from a website page, remove any HTML code, and preserve the data. It is a web scraping tool that helps to organise and analyze the files downloaded from the Internet to extract names and URLs from an HTML page.) It allowed the author to quickly retrieve the needed data by offering straightforward techniques for navigating, and searching a website.

3.3.2 URL Classifier (EGClass)

The URL classifier module was built using Python programming language. The classifier, EgClass is able to distinguish between a good and malicious website by using the Machine Learning algorithms and a dataset which consist of features that predicts the likelihood of a website being malicious. EgClass makes use of the developed Model and the vectorizer to classify the links into malicious or benign. The function of the vectorizer is to convert the

contents of the webpage into numeric data that the clarifier can understand. The URL classifier classifies the contents of the CrawledLinks and temporary keep them in the Classified malicious file where the information will be used by GUI to update the database. In the Figures 3.3 and 3.4 which clearly show the flow for the URL classification architecture. The Support Vector machine was used for the classification with an accuracy rate of 92.75%.

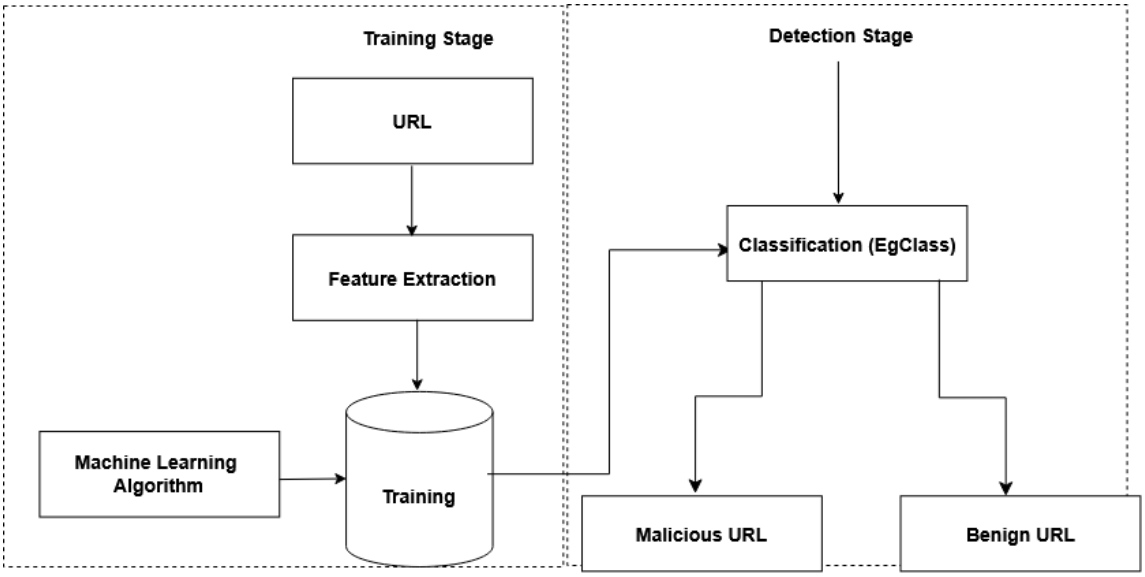


Figure 3.3: Malicious URL Detection and classification Model using ML

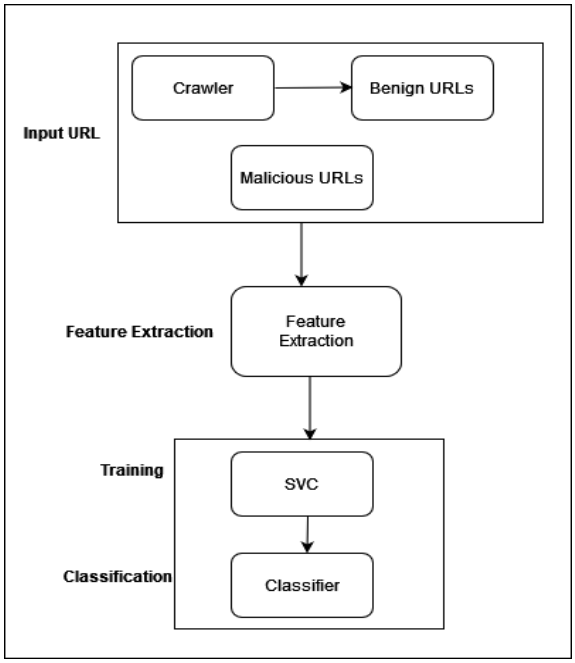


Figure 3.4: URL Classification Architecture (Mamun et al., 2016)

3.3.3 Machine Learning Module

This module integrates machine learning models for the URL classification and threat detection which handles phases like the Model training and Model evaluation. It also incorporates feature engineering, model selection, and performance optimization techniques. A total of 3743 URLs were analysed in this study; 80% was used for model training, while the remaining 20% of the dataset was used for testing and validation of the model. The dataset can be separated into two sections (Figure 3.5): the train/test split, where the model is built using the training dataset and its predictive power is evaluated using the testing dataset (Nguyen et al., 2021).

Partitioning data into subsets for independent model training and evaluation is known as data splitting (Joseph & Vakayil, 2022) . The process of dataset splitting is seen to be essential and crucial for removing or minimising bias in training data for machine learning models (Muraina, 2022). The aim of splitting the dataset is to prevent over-fitting and bias in model selection; the testing dataset can share the same percentages with the cross-validation set, which is the development set, and the training set should be the largest (Ahsan et al., 2021). Birba (2020) similarly stressed that the model can be trained by cross-validation once the training set has been split up into several sets. Since separating datasets into training and testing datasets is a difficult process, the training set should be subjected to frequent application on the training dataset.

Although dividing a dataset into a number of categories may appear simple, researchers must proceed carefully because the size of the datasets and the train/test split ratios can have a significant impact on the models' results as well as the classification performance itself. To divide data into a train, test, or validation set, data splitting is therefore frequently employed in machine learning. By using this splitting technique, the researcher is able to measure the generalisation performance and identify the model hyper-parameter (Muraina, 2022).

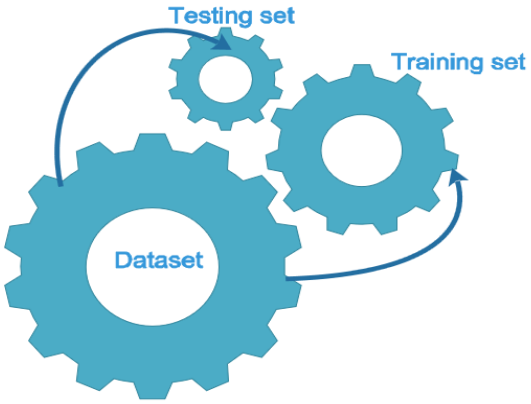


Figure 3.5: Dataset Split

The HTTPS status, the number of dots in the domain name, and a few top-level domain-related attributes are some of key input data points that were employed in the analysis. Three machine learning methods, including support vector machines (SVM), decision trees (DT), and random forests (RF), were employed in this study. With an accuracy rate of 92.75%, the results showed how well the SVM model performed, and as a result, it was chosen as the best model. Figure 3.6 shows the flowchart of the Support Vector Machine used in this study.

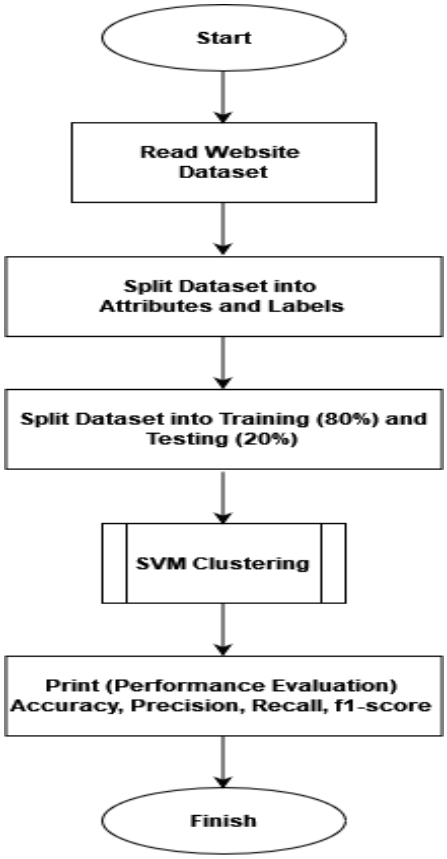


Figure 3.6: Support Vector Machine Flowchart

The following are the steps involved for classifying dataset with the SVM algorithm.

a. Import the dataset of malicious websites.
b. Label and separate the data into attributes
c. Separate the data set into two sections: 20% for testing and 80% for training.
d. Following data pre-processing, the Support Vector Machine (SVM) technique was used to classify the data
e. Print Accuracy, Precision, Recall, and f1-score from data testing.

3.3.4 Database Management Module (The Rule Manager)

A rule manager system was engineered using the Python programming language, leveraging the PyCharm Integrated Development Environment (IDE) for its creation. The development of this database was meticulously executed using Microsoft Visual Studio Code. The browser extension was encapsulated within a manifest.json file, which explicitly conveys the extension's structure. It also includes an index.html file that serves as the user interface component, designed to display appropriate information when a blacklisted site is accessed by the user. Upon visiting a

website, the extension vigilantly scrutinizes the URL against the entries stored in the database. If a match is detected, the extension promptly presents the user with the associated index.html content. Furthermore, it can be used to detect malicious websites and the user is warned about the risk before visiting such website.

This Rule Manager was ingeniously designed to provide a comprehensive suite of functionalities, comprising:

i. Add Rules: This feature allows for the addition of new website addresses to the rule.json database, following a predefined and standardized format. Before appending a new entry, the system diligently verifies whether the website URL already exists within the database. If the URL is not present, it is duly incorporated, and the system intelligently calculates the index for the new entry.
ii. Delete Rule: In circumstances where the need arises to expunge a URL from the blacklist, the Delete Rule functionality efficiently carries out this task. It rigorously validates the existence of the rule prior to initiating the deletion process.
iii. Check Rule: This function act as a vigilant sentinel, verifying the presence or absence of a specific rule, offering essential support to other Rule Manager functionalities.
iv. Count Rule: By employing this feature, the total number of entries contained within the blacklist is determined, providing valuable insights into the scope of the database.
v. Rule Map Builder: This ingenious functionality constructs the rule database into a structured and searchable map object. This transformation significantly enhances the efficiency of searching and retrieval processes within the database.

The Rule-based Manager with JSON file was used as against the traditional relational database management system because of its simplicity, portability, space efficiency, improved query performance, efficient storage and retrieval and high degree of customisation and flexibility in the definition of data rules and structures.

Technology used for developing the Rule Manager

The filter relies on a database known as ‘Rules’, which is represented as a structured JSON file. Within this database, websites are categorized as either malicious or benign based on their content. Each malicious website entry adheres to a specified JSON object format, exemplified in the Figure 3.7.

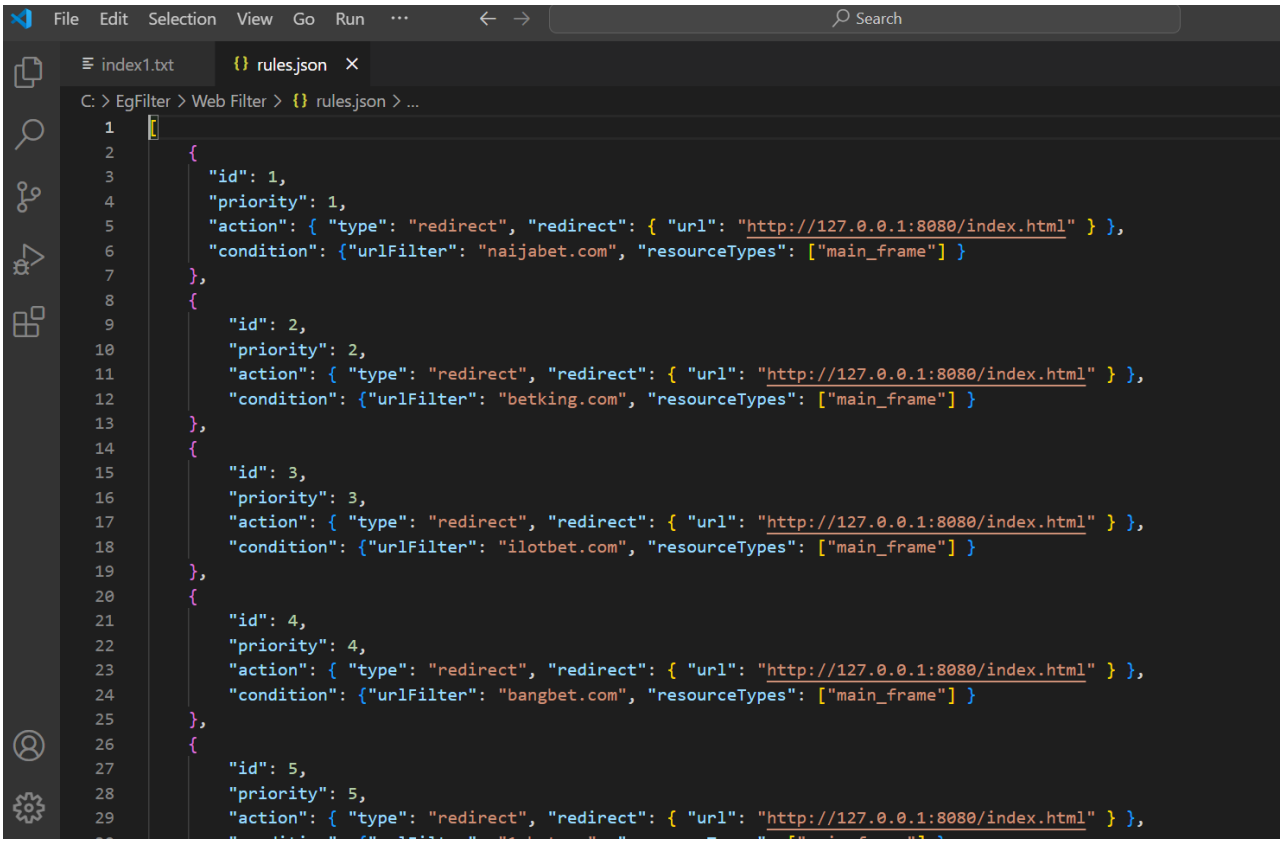


Figure 3.7: The index.html file displayed on the client side

JSON is a lightweight data interchange format meant for humans to easily read and write, and also easy for machines to parse and generate; which makes it a popular option for data exchange and storage It is also used for the transmission of data between a server and a web application as a substitute to XML (Rask et al., 2020).

3.3.5 Egenti-Filter's Graphical User Interface (GUI)

The development of the Egenti-Filter's Graphical User Interface (GUI) was accurately implemented employing the C# programming language within the Microsoft Visual Studio Integrated Development Environment (IDE). The GUI, stands as a vital component of the research's implementation and offers prospective users the following user-friendly options to manage, update and analyse the rule database:

- Update the Database: This option empowers users to seamlessly update the database with newly classified malicious URLs, as identified by the model.
- Manual Addition of URLs: Users are equipped with the capability to manually append a known malicious URL to the database, ensuring that the database remains updated.

- Manual URL Removal: Should the necessity arise to expunge a URL that is no longer deemed malicious or has been erroneously classified, the GUI includes a feature for the manual removal of URLs.
- Database Analysis: The GUI endows users with the capacity to perform diverse database operations, including counting records, conducting descriptive statistics, and undertaking various analytical tasks to glean insights.
- Search Functionality: A robust search function has been meticulously integrated into the GUI, enabling users to effortlessly search for specific URLs within the database. This search capability enhances user convenience and expedites retrieval of critical information.

1. **Frontend (GUI) Technologies:**

   C#:

   - **Description**: C# was chosen for developing the graphical user interface (GUI) of the web filter extension due to its robustness, performance, and compatibility with the .NET Framework.
   - **Use Cases**:
     - GUI Design: Designed user-friendly interfaces using C# and XAML for elements like buttons, text boxes, lists, and menus.
     - Event Handling: Implemented event handlers and logic for GUI interactions, such as adding URLs to the blacklist, triggering model updates, and displaying classification results.
     - Integration with Backend: Established communication with backend APIs using HTTP requests (e.g., GET, POST) for data retrieval and processing.
     -

2. **.NET Framework**:
   - **Description**: The .NET Framework provided a comprehensive platform for C# application development, offering libraries, tools, and runtime support for building Windows applications.
   - **Use Cases**:
     - Framework Components: Leveraged .NET libraries for tasks like file I/O, data binding, serialization, and GUI controls.

- Compatibility: Ensured compatibility and seamless integration with Windows operating systems for deploying the web filter extension.

3.3.6 Egenti-Filter: Web Browser Extension Framework

The designed research framework for the encapsulate Egenti-Filter captures the stages needed to develop an improved web browser extension filtering model for secure online activities using web content filtering approach. Egenti-Filter framework was developed to drive this research objective. The framework is a group of components of an online Web Browser Extension capable of securing users' activities. As shown in Figure 3.8, the components include: Graphical User Interface (EgFilGUI), Web crawler (EgCrawl), Temporary DB (fetched URLs), URL classifier (EgClass), Temporary DB (Classified URLs), EgFilter (Egenti-Filter), Blacklist DB, Browser, and Filtering Navigation. The following subsections describe the framework with the different stages.

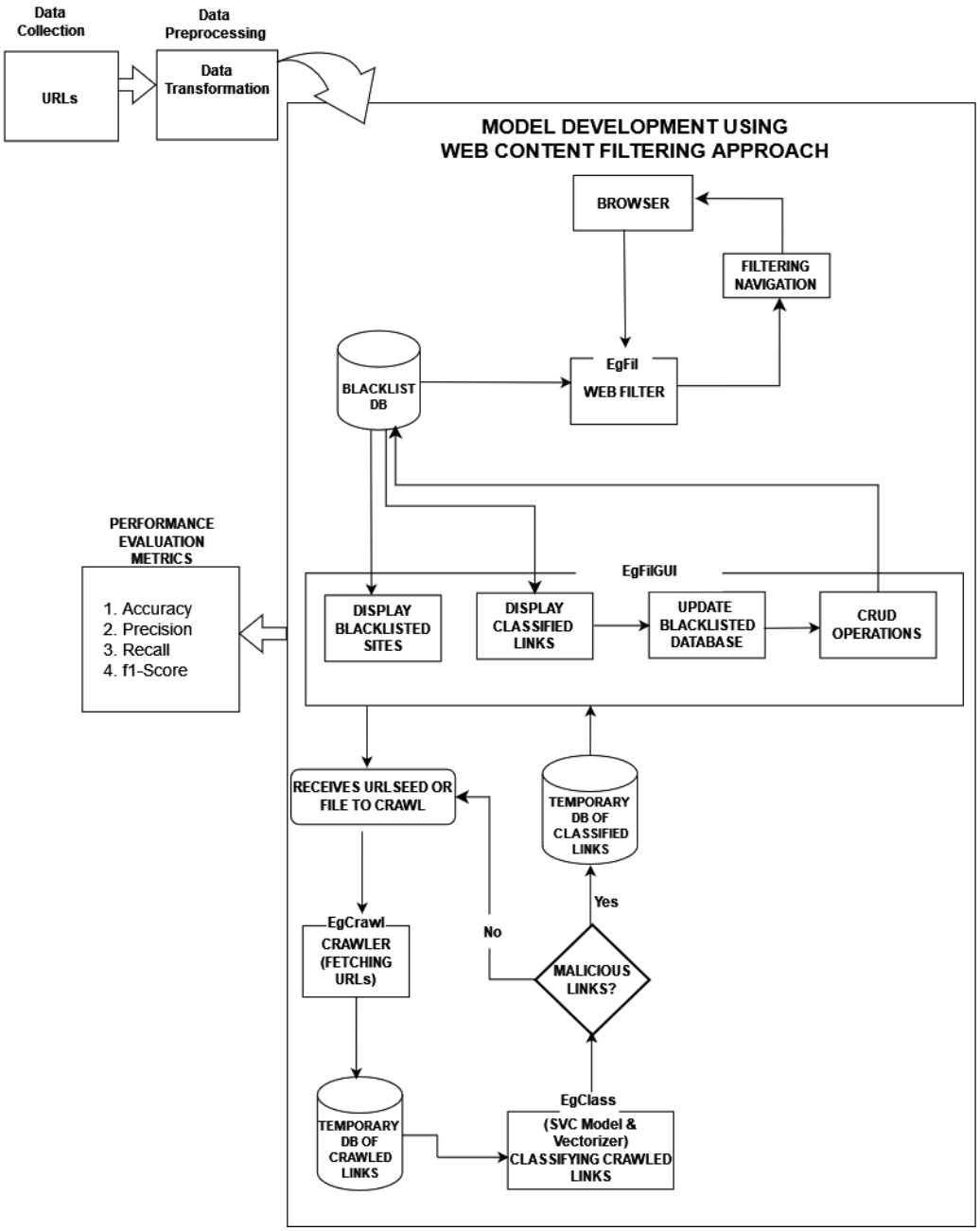


Figure 3.8: Egenti-Filter (Web Browser Extension) Framework

i. Data Collection

The initiation of this research endeavour commenced with acquisition of a diverse dataset. Multiple sources were tapped, including data from google.com, yahoo.com, whois.com, virusTotal application, and the web crawler. This data compilation encompassed a wide array of cases pertinent to the research problem, resulting in a comprehensive dataset comprising 3743 records, which was divided into two parts that were used for both the training and testing data,

which resulted in 2,994 training data and 749 for testing, representing 80% and 20% respectively. A sample of the data collected for this research is shown in the Figure 3.9. These URLs were checked by VirusTotal tool to authenticate labels of each identified URL. The complete dataset was stored using the CSV format. For each URL sample, there is a label- 'bad' for malicious and 'good' for benign website. The features that were extracted are the length of the URL, the number of special characters, the site age, the country and the operating system from which the website was fetched, and these variables have been helpful in identifying malicious websites. PyCharm with Scikit-learn python library was used in the processing of this data to analyze the classification result.

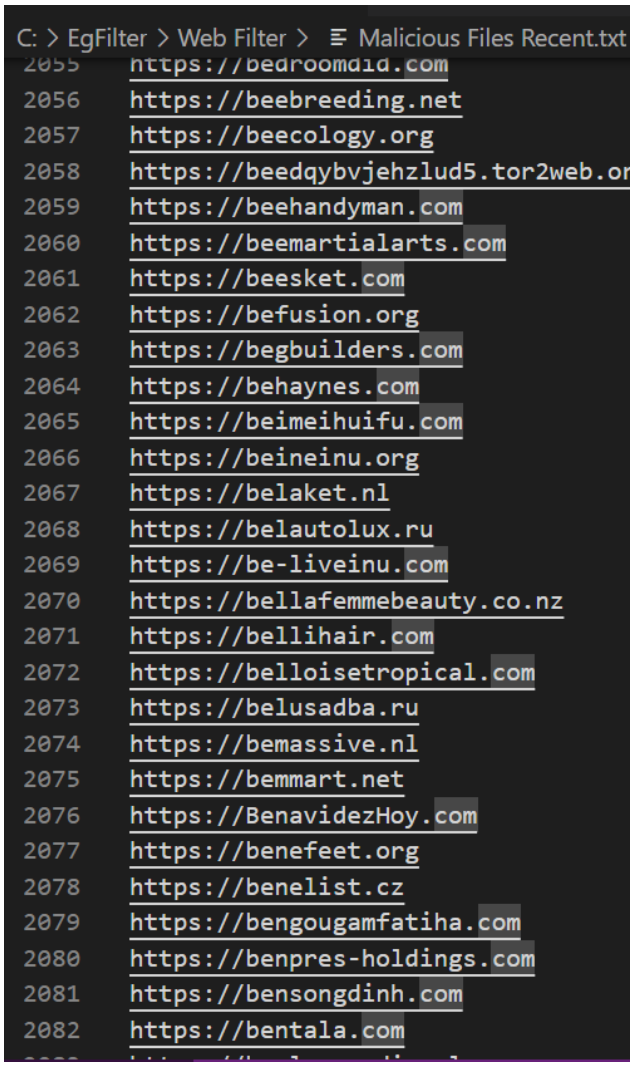


Figure 3.9: Sample dataset used for the Study

ii Data Preprocessing

Following the successful acquisition of data, rigorous preprocessing measures were implemented to fortify the dataset's quality and suitability for training algorithms; hence preparing the dataset for machine learning algorithms. These preparatory measures included comprehensive data cleaning, meticulous handling of missing values, and judicious feature scaling. Additionally, a comprehensive Exploratory Data Analysis (EDA) was conducted to gain profound insights into the intrinsic characteristics of the dataset, and it examines the datasets prior to using machine learning algorithms. EDA is essential for knowing the properties of the dataset and getting it ready for classification tasks. To sum up, EDA was an essential phase in the data preprocessing for this project since it involves web classification. In order to create precise and useful classification models for the dataset, the author investigated, cleaned, and prepared the dataset, found out important features, dealt with outliers, comprehended correlations, and visualized data relationships. For example, the EDA helped to know if the dataset was balanced or imbalance, the number of data in the dataset, and if there were outliers or not.

Technologies Used for the Web Browser Extension

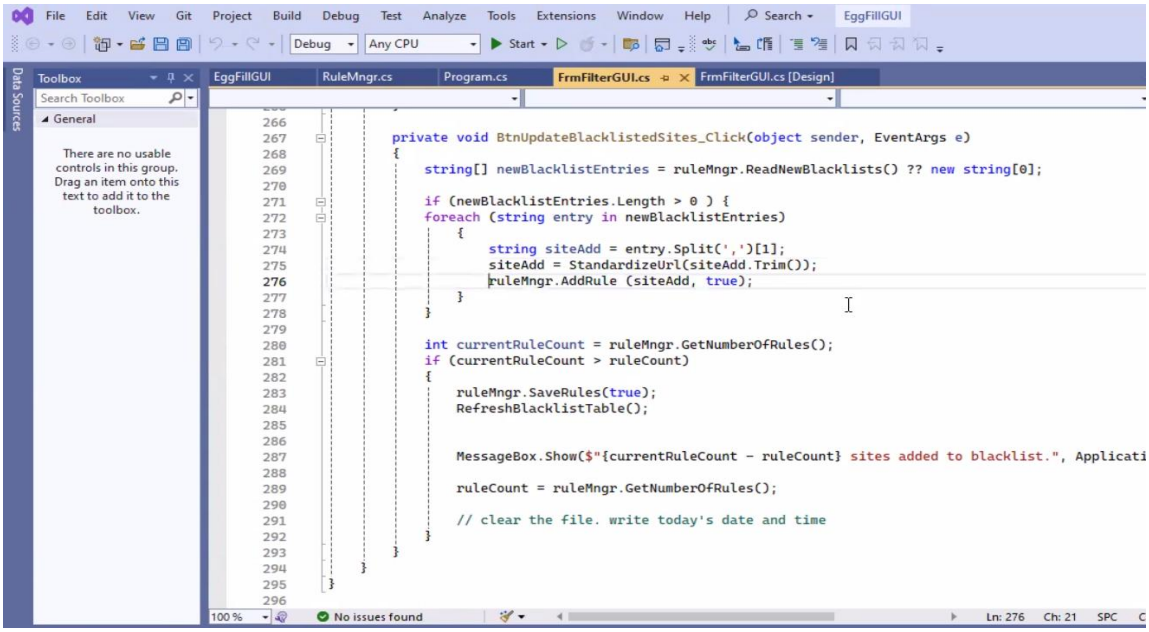
1. **Python**:
    - **Description**: Python was chosen as the primary backend scripting language due to its versatility, extensive libraries, and ease of use for web-related tasks.
    - **Use Cases**:
        - Web Crawler: Utilized libraries such as the Requests and BeautifulSoup for web crawling and data extraction from URLs.
        - Model Training: Integrated with Scikit-learn to train machine learning models for URL classification.
        - API Development: Developed RESTful APIs using Flask for communication between the frontend GUI and backend functionalities.
        - Data Management: Handled data processing, rule management, and database interactions using Python scripts.
2. **Scikit-learn**:
    - **Description**: Scikit-learn, a popular machine learning library in Python were used for the model training, evaluation, and selection.
    - **Use Cases**:

- Model Training: Implemented algorithms like Random Forest, Support Vector Machine (SVM), and Decision Tree for URL classification.
- Evaluation Metrics: Calculated evaluation metrics (e.g., accuracy, precision, recall) using Scikit-learn functions to assess model performance.
- Hyperparameter Tuning: Conducted hyperparameter tuning using techniques such as grid search or random search for optimizing model performance.

3. **JSON**:
    - **Description**: JSON (JavaScript Object Notation) served as the database format for storing and managing classified URLs, rules, and other relevant data.
    - **Use Cases**:
        - Database Storage: Saved blacklisted URLs, model parameters, and classification results in JSON format for easy retrieval and manipulation.
        - Data Serialization: Serialized and deserialized Python objects to JSON format for efficient data exchange between backend components.

3.3.7 Instating Egenti-Filter as a Web Browser Extension

Implementing the designed web content filtering framework involves developing the web content filter (Egenti-Filter), developing the filter as a browser extension and also developing a graphical user interface for easy interaction with the system. In the development of this research work, Egenti-Filter, which is the brainbox of this research endeavour; JavaScript, Hypertext Markup Language (HTML), and Cascading Style Sheet (CSS) were the programming languages that were used. JavaScript is a computer language that is largely utilized by Web browsers to give users an engaging and dynamic experience. Javascript is widely used for frontend development, required for integrating the filtering system into web browsers or web-based applications. A snippet of the codes are displayed in the Figure 3.10.



1Figure 3.10: Snippet of Python codes

Markup language refers to HTML's usage of tags to distinguish between various content kinds and the tasks they each provide to the webpage, as opposed to a programming language. Regardless of the intricacy of a site or the number of technologies included, HTML is the foundation of every online page. Figure 3.11 which shows the IDE for writing HTML tags. HTML especially makes use of tags, sometimes called elements. These HTML tags, which indicate the many types of material on a page, make up every web page. Every kind of material on the website is encircled by HTML tags.

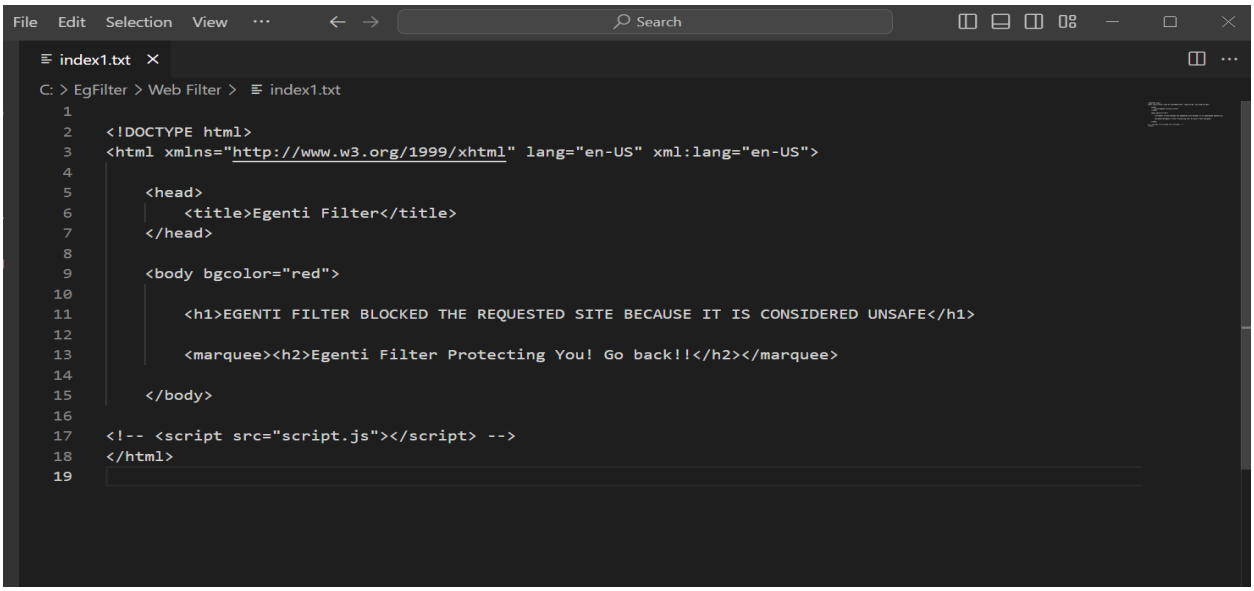


Figure 3.11: HTML Index File

CSS is a programming language that specifies how HTML elements should actually look on a page. While HTML offers the basic tools for organising content on a website, CSS helped in styling the content so that it looks as intended to the author (Rask et al., 2020). Example of the CSS codes used for styling the index.html file in this project is shown in the Figure 3.12, styling the Graphical User Interface to look attractive to the eyes. These languages are kept apart to give websites the proper structure before they are reformatted. These programming tools were all combined to give the GUI the look and feel that it has.

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" lang="en-US" xml:lang="en-US">

    <head>
        <title>Egenti Filter</title>
    </head>

    <body bgcolor="red">

        <h1>EGENTI FILTER BLOCKED THE REQUESTED SITE BECAUSE IT IS CONSIDERED UNSAFE</h1>

        <marquee><h2>Egenti Filter Protecting You! Go back!!</h2></marquee>

    </body>

<!-- <script src="script.js"></script> -->
</html>
```

Figure 3.12: CSS Styling Tags

Python is a high-level programming language with an extensive standard library, object-oriented programming, and fundamental, practical programming (Jijo & Abdulazeez, 2021). Using Python, a language renowned for its ease of use and potent web data processing features, and the extensive library ecosystem makes it a great platform why it was considered by the researcher to use it in developing the web crawler application for this research project. Some of the project component; EggFill.py is the Model itself, the Util.py is for data cleaning, Site_address.csv is the data as displayed in the Figure 3.13.

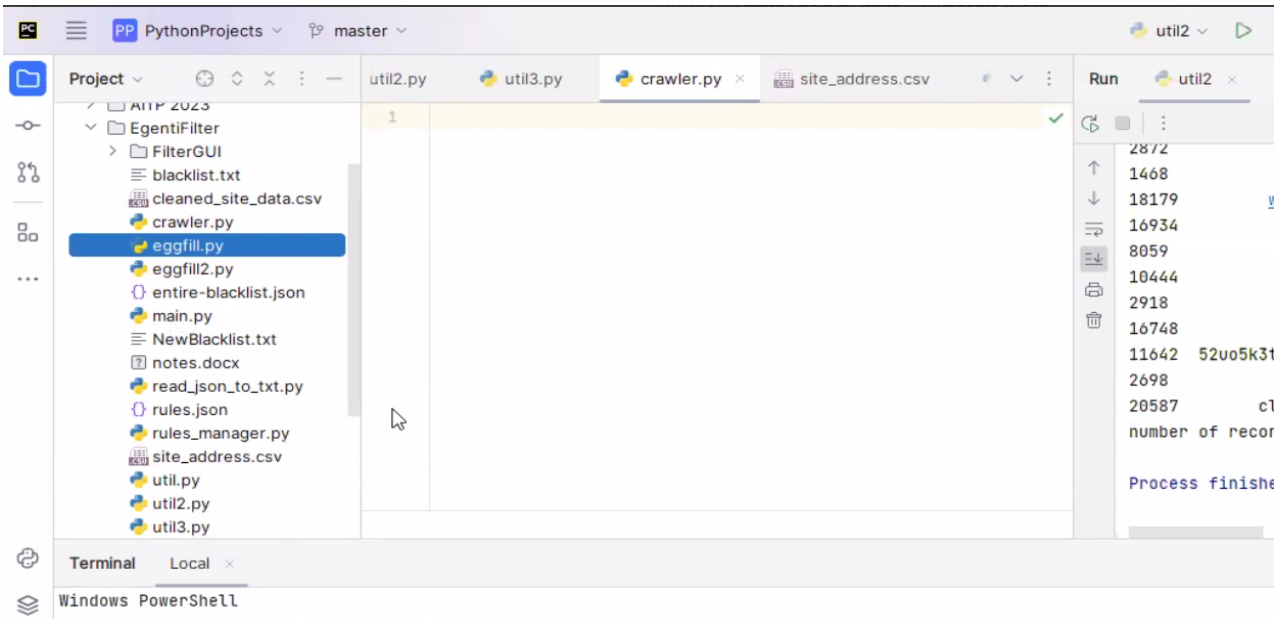


Figure 3.13: Python interface showing some components of the project

3.3.8 Employ Python Library (scikit-learn) for model performance

A comprehensive model evaluation was undertaken to rigorously assess the performance of the trained Web Filtering Model. The evaluation process entailed the deployment of key metrics, including: Accuracy, Precision, Recall, and f1-score.

a). Accuracy: Accuracy depicts the proportion of correct predictions over the total number of predictions. It is represented as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TN} + \text{TP} + \text{FP} + \text{FN}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad 3.1$$

A higher degree of accuracy signifies that the system exhibits superior ability in differentiating between malicious and benign or legitimate URLs, thereby diminishing the likelihood of false positives and false negatives

b). Precision: This measures the proportion of correctly identified malicious URLs out of all URLs classified as malicious by the model. It is represented as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad 3.2$$

A high precision number signifies that the model's ability to accurately classify a URL as malicious is high, hence reducing the likelihood of erroneously categorizing real URLs as malicious.

c). Recall: This evaluation parameter also known as True Positive Rate (TPR) or Sensitivity measures the ability of a model to correctly identify all actual malicious URLs out of all malicious URLs present in the dataset. It is given as:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{3.3}$$

A high recall score signifies that the model is proficient in identifying the majority of malicious URLs in the dataset, hence minimizing the likelihood of overlooking genuine malicious attempts.

d). F1-score: This represents the harmonic mean of the Precision and the Recall parameters and it is given as:

$$\text{f1} - \text{Score} = 2 \text{ x } \left(\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}\right) \quad \text{3.4}$$

A high f1-score signifies that the model attains a combination of impressive precision (with minimal false positives) and recall (with minimal false negatives), rendering it a dependable indicator of the classification system's overall efficacy in detecting malicious URLs while minimizing misclassifications.

Where:

True Positives (TP) - refers to a situation when both actual class and the predicted class of data point is 1. In essence, TP count of malicious websites is the malicious URLs that are correctly predicted as malicious

True Negatives (TN) - refers to a situation when both the actual class and the predicted lass of data point is 0. In actual sense, TN count of benign URLs that are correctly predicted as benign.

False Positives (FP) - when the actual class of data point is 0 and the predicted class of data point is 1. This reflects the count of benign URLs that are incorrectly predicted as malicious.

False Negatives (FN) – when the actual class of data point is 1 and the predicated class of data point is 0. This depicts the count of malicious URLs that are incorrectly predicted as benign.

Technologies used for Performance Metrics Computation

The Python script for computing performance metrics was chosen and implemented using specific technologies for the following reasons:

Python Programming Language:

- Utilized for script development, defining functions, and implementing algorithms for computing performance metrics.
- Selected for its simplicity, readability, and extensive support for scientific computing and data analysis tasks.
- Offers a wide range of libraries (e.g., NumPy) that simplify complex mathematical computations and statistical operations.

NumPy Library:

- Leveraged for numerical calculations, array-based operations, and statistical computations needed to derive performance metrics from input data.
- Utilized for efficient numerical computations, array manipulation, and mathematical operations required for performance metric calculations.
- Provides optimized functions for handling arrays and matrices, enhancing computational speed and accuracy.

3.3.9 Model Training and Evaluation

1. **Data Preparation**:
    - Data Collection: Gather labeled data (malicious URLs, safe URLs) from sources like threat intelligence feeds, public datasets, user feedback, Search engines.
    - Data Labeling: Manually or programmatically label URLs as malicious or safe, ensuring a balanced dataset for training and evaluation.
2. **Model Training**:
    - Feature Engineering: Extract features from URLs (domain, path, query parameters), page content, and metadata using techniques like TF-IDF, word embeddings (Word2Vec, GloVe), and n-grams.
    - Model Selection: The author used different algorithms (Random Forest, SVM, Random Forest, and Decision Tree) and Python programming language using

libraries like scikit-learn until Support vector classifier was selected for the best performance accuracy of 92.75%.

- Hyperparameter Tuning: Fine-tune model hyperparameters using techniques such as GridSearchCV, RandomizedSearchCV, and Bayesian optimization to improve model performance.

3. **Model Evaluation**:
    - **Performance Metrics**: Evaluate models using metrics like accuracy, precision, recall, F1-score to assess classification performance, model robustness, and generalization capability.
    - **Cross-Validation**: Perform k-fold cross-validation to validate model stability, prevent over-fitting, and estimate model variance across different subsets of the dataset.

3.3.10 Implementation Details

Backend Modules

1. **Rule Manager**:
    - **Rule Database**: Implemented using a JSON file. The database schema includes fields for blacklist rules, whitelist rules, configuration settings, and rule metadata.
    - **Rule Management API**: Developed using Python and a web framework, Flask was used. The API exposes endpoints for CRUD operations on rules, rule updates based on user actions, and rule application during URL classification.
2. **Data Collection and Preprocessing**:
    - **Web Crawler/Scraper**: Built using Python libraries like The Requests and Beautiful Soup. The crawler fetches URLs, extracts metadata (title, description), and preprocesses web content by removing HTML tags, tokenizing text, and filtering out irrelevant information.
    - **Preprocessing Pipeline**: Includes Python scripts for data cleaning, normalization (lowercasing, removing stop-words), feature extraction (TF-IDF, word embeddings), and vectorization (converting text data into numerical format) using the scikit-learn libraries.

3. **Machine Learning Model**:
    - **Model Selection**: Python code for selecting the machine learning model based on performance metrics, dataset characteristics, and domain requirements. Common models include Random Forest, Support Vector Machine (SVM), Logistic Regression, or deep learning models like LSTM or CNNs for sequence data. But the Support Vector machine was used for this project.
    - **Training Pipeline**: Used scikit-learn for model training, hyperparameter tuning (GridSearchCV, RandomizedSearchCV), cross-validation (k-fold validation), and model evaluation using metrics like accuracy, precision, recall, and F1-score.
    - **Computer system**: Core i5, 8GHz Personal Computer (PC) was used to carry out all the software operations
    - **Draw.io:** This software was used to design the flowcharts, framework, and diagrams.

Frontend (GUI):

1. **Browser Extension User Interface**:
    - **Popup Interface**: Developed using HTML, CSS, and JavaScript (often with frameworks like React, Vue.js, or Angular) for creating interactive User Interface elements in the browser extension popup.
    - **User Settings**: Allows users to configure extension settings, manage blacklists/whitelists, view classification results, provide feedback on URLs, and access help/documentation.
2. **Warning Page**:
    - **HTML Templates**: Custom HTML templates with CSS styling for displaying warning pages when users visit malicious URLs. These pages may include warning messages, risk indicators, and actions for users to take (e.g., go back, report as safe/malicious). In this study, the developed web browser extension displayed a warning page whenever the user tries to visit any of the malicious websites.

3.4 Questionnaire Administration

i. Introduction

This section presents the first research methodology used in this study. The research design, population, sampling technique, data collection methods, and data analysis procedures are presented in detail. The primary data collection instrument used in this study was a questionnaire, known as Egenti-Filter Securing online activities (Egenti-Filter SOA Survey), which was designed to gather quantitative data from cybersecurity professionals regarding their awareness on web content filtering and practices concerning online security.

ii. Research Design

This study adopts a purposeful research design to assess the level of awareness among cybersecurity practitioners on the use and adoption of web content filters, the different types of web filters, its effectiveness, limitations, and the current security measures on social networking sites. The choice of this design is based on its ability to gather detailed, structured data from a target sample of respondents.

iii. Population of the Study

The target population comprises of individuals aged 25-64 years who are experienced in the IT field, especially in cybersecurity and related fields, either by the nature of their job or by social research association to get the best responses which proves the integrity of the data collated. This population is selected based on the assumption that they are familiar with the concept of web content filtering and are likely to have some level of experience with online security concerns. It assess the level of their awareness on the web content filtering systems, the effectiveness of existing filtering techniques, the limitations of existing filtering techniques, among other question items.

iv. Sampling Technique

A purposive sampling technique was used to select a sample of 162 participants who meet specific criteria:

- Regularly users web browsers
- Knowledgeable about web content filtering

- Knowledgeable about online security threats
- Experienced with browser extensions.

v. Data Collection Instrument

The primary instrument for data collection was a structured questionnaire, designed to assess cybersecurity practitioners' perception on the effectiveness, ease of use, and overall impact of the web content filters. The questionnaire also aimed to gather data on respondents' awareness on current security control measures on social networking sites. The questionnaire was divided into nine sections:

1. **Section A:** Demographics
2. **Section B:** Awareness of Internet Filtering
3. **Section C:** Challenges and concerns of Internet Filters
4. **Section D:** Internet Filtering Tools and technologies
5. **Section E:** Limitations of web content Filters
6. **Section F:** Overall effectiveness of web content filters
7. **Section G:** Awareness of data collection
8. **Section H:** Appraising the current security and control measures
9. **Section I:** Improvement Suggestions

vi. Validity and Reliability of the Questionnaire

To ensure the validity of the questionnaire, a pilot study was also conducted with a small group of 23 participants, which cuts across friends, family and colleagues. Feedback was gathered to ensure that the questions were clear, relevant, and captured the intended data, especially as it concerns the research questions for this study. Minor revisions were made to improve the clarity of the questions based on this feedback.

vii. Ethical Considerations

i. In line with NOUN's research ethics policy, an ethical clearance was formally obtained from ACETEL (Appendix A).

ii. All prospective participants were presented with an informed consent comprising of the right to participate, the right to withdraw at any point, the right to anonymity, and the right to confidentiality of responses before attempting the Questionnaire. Furthermore, participants' participation was optional and voluntary (Appendix B).

iii. The Questionnaire (Appendix C) items were presented to the chief supervisor for correctness and validation

viii. Questionnaire Distribution

The Questionnaire was designed in Google Form and distributed through the Internet, via email, social medial platforms and professional networks; this is to aid a wider coverage to as many respondents as possible. For example, the research engaged different media, platforms and avenues in getting the Questionnaire across to the prospective respondents, for instance, it was distributed through the author's Facebook account, LinkedIn account, and different LinkedIn groups, e.g., cybersecurity forum initiative (CSFI), and it was also distributed to some cybersecurity professionals LinkedIn accounts. The Questionnaire was also distributed to some Whatsapp groups, e.g., Cyber Security Experts Association of Nigeria (CSEAN), InfoTech News Hauz, Nigeria Computer Society (NCS), Abuja Chapter, and the Directorate of Information and Communications Technology (DICT) of the National Open University of Nigeria (NOUN). The author further engaged calls, SMSs and personal Whatsapp messages to individuals. Functional email addresses of cybersecurity professionals were obtained from different sources, e.g. Cybersecurity mailing lists from workshops and conferences. Friends, family and colleagues also aided in the sharing of the Questionnaire link; this is to encourage prospective respondents to participate in the survey. Participants were given four weeks to respond to the questionnaire, with a reminder sent at the end of every week to ensure adequate participation.

ix. Data Analysis

The data collected through the Questionnaire was analysed according to the specific research question being addressed; percentages were used to analyse the research questions. The Age percentage of responses and the number of years of expertise in the cybersecurity sphere were considered, some of the items in the Questionnaire were the bedrock of this research. Presenting the processed data using the data visualization software known as Tableau was the last stage of the workflow. Tableau facilitates the presentation of data in an aesthetically pleasing manner,

hence facilitating stakeholders' comprehension and interpretation of the insights gleaned from the data. Tableau is renowned for its user-friendly interface and its capacity to produce dynamic, eye-catching data visualizations (graphs and charts, such as scatter plots, line charts, and bar chart), report generation and interactive dashboards, and geographical visualizations and maps.

CHAPTER FOUR

RESULTS

4.1 Preamble

This chapter presents the graphical, tabular and narrative information of the research findings, and summarizes the results obtained as related to the study objectives (section 1.4). The research goal (section 1.3) is centred on model development and the distribution of the research instrument - Questionnaire (Appendix C) towards securing online activities. The following sections depicts the outcome of various research techniques adopted on the research methods as fully discussed in chapter three.

4.2 Ethical Considerations

This research addresses the ethical approval and due diligence as follows:

i. For the model development, unwavering commitment to ethical principles was maintained. This encompassed a steadfast dedication to data privacy and a vigilant approach to bias mitigation. These ethical considerations were upheld to the highest standard in both the development and deployment of **Egenti-Filter** as the overarching commitment to ethical conduct is integral to this research endeavour

ii. The author subjected the entire thesis document to thorough language editing and correctness, ensuring linguistic and technical accuracy by reducing and eliminating grammatical, syntactical and spelling errors. The certificate is as depicted in (Appendix D).

iii. In compliance with data protection and privacy requirements (DPPR), the author ensured and will continue to ensure compliance with browser store guidelines. The users have the right to complain if browser extension performs below the expected, and we shall continuously seek user feedback on privacy and security issues with our extension. An email address will be provided alongside the documentation. The author is to perform regular privacy impact assessment to identify and assess potential risks to user, and proactively mitigate the identified risk, and stay updated on changing privacy laws and regulations. There shall not be non-disclosure of users' data to third party organisations with the user's consent. User data shall be anonymised to protect user's identities in cases of data breach. Finally, the author ensured that the extension is free of any security vulnerabilities such as cross site scripting (XSS) and cross site request forgery (CSRF).

4.3 Questionnaire Analysis

The analysis of the Questionnaire shows a significant relationship between online security and web content filtering, that the approach of using web content filters serve as an effective cybersecurity tool to prevent cyber-attacks from infiltrating personal computers, organisations' network, or school library systems. Figure 4.1 shows the reporting dashboard using Tableau – a data analysis tool – to analyzing participants' responses as follows:

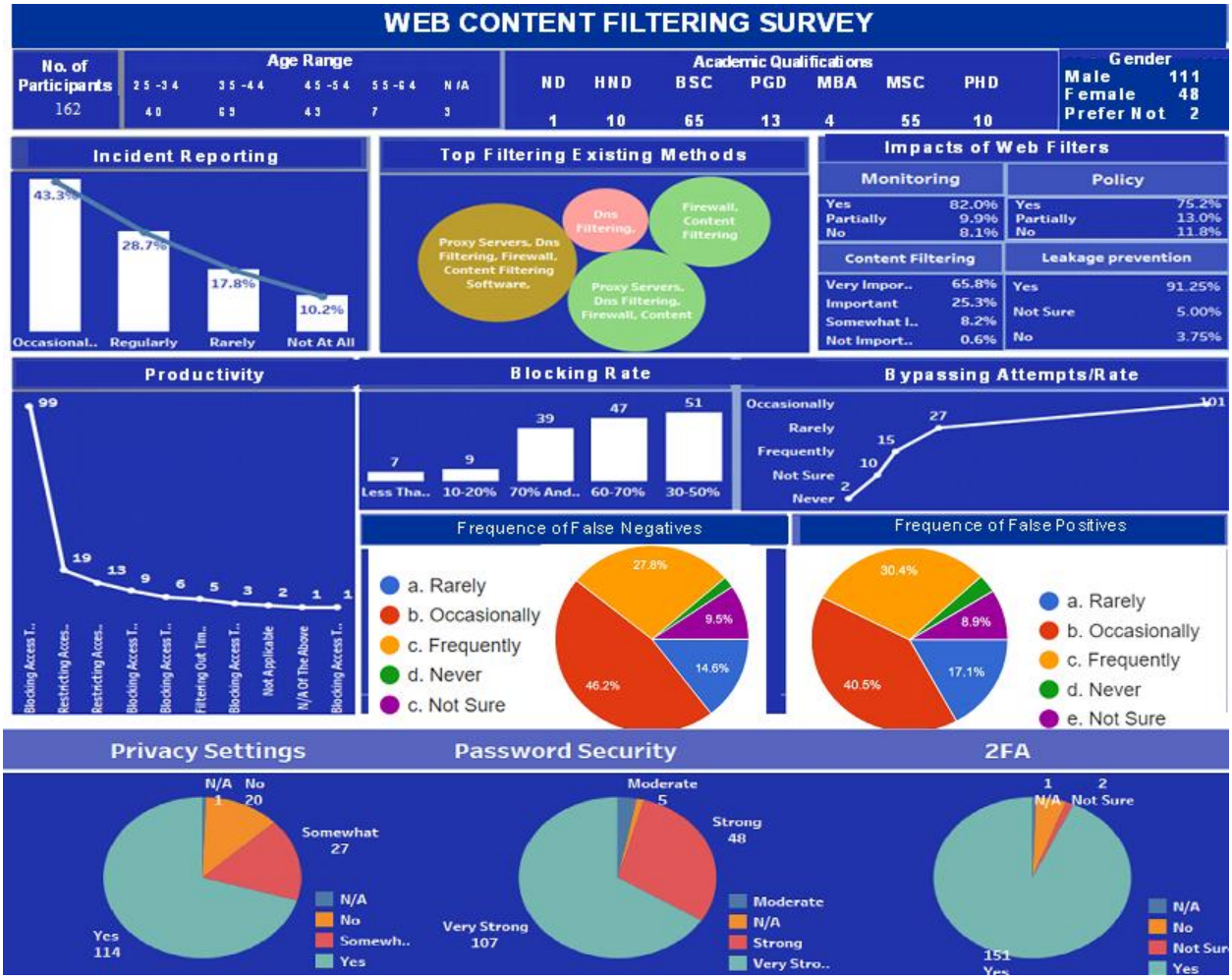


Figure 4.1: Dashboard Reporting System for Egenti-Filter SOA Survey

i. The demographics for the survey are provided on the upper side of the chart showing the participants' age range, gender and qualifications.

ii. Responses collected show low attention to activities reporting (false positives and false negatives). Hence, more users' awareness on the importance of security incidents' reporting is crucial.

iii. The top filtering approaches were recorded as (DNS Filtering, Firewall, content filtering software, and proxy server).

4.3.1 Analysis of the Questionnaire

This questionnaire design is targeted at identifying the most common web filtering systems, and to identify the effectiveness of the existing web content filtering systems. The design also aims at ascertaining the users' awareness levels of the web content filtering systems, as well confirming its importance.

4.3.2 The Common Web Filtering Solutions

The top on the list is Firewalls (87.2%) and URL Filtering (87.2%), followed by Content Filtering Software (71.8%), and whitelisting and blacklisting of IP address (63.5%), while keyword-based filtering recorded 63.5%, as depicted in the section labeled as 'Top filtering existing methods' in Figure 4.1.

4.3.3 Effectiveness of the Existing Web Filters

To measure the effectiveness of existing web filters, the questionnaire tested the following features: the blocking rates, bypassing attempts, and the false positives and false negatives (Figure 4.1).

i. **Blocking Rates** - This has to do with the percentage of inappropriate or unwanted content that are successfully blocked by the filtering measure. 'Blocking Rate' section shows that web filters only block a total of 30-50% of inappropriate contents, calling for improvements on the proposed filtering model for this study. Web filters face significant challenges in attaining 100% accuracy as a result of the rapid growth of the Internet (ITU, 2020); with approximately 175 new websites created every minute (StatsFind, 2022), amounting to roughly 252,000 new websites created daily around the world, it is inevitable that some of these will contain offensive or malicious content.

ii. **Bypassing Attempts/Rates:** These are question items that determine how often the attempts to bypass the filtering measures. Nash (2021) investigated the effectiveness of internet filtering technology on the adolescents web experiences to confirm that the technical aptitude of social network users varies, and those tech-savvy individuals are more likely to circumvent filtering technologies. Consequently, the study asserted that the

efficacy of such filters in protecting users from harmful online content may be limited for the group of users with high technical skills. The 'Bypassing Attempts/Rate' shows the rate of bypassing attempts by users. Though 'occasionally' has the highest attempt; however, the survey did not probe further if there was successful attempts or not.

iii. **False Positives and False Negatives:** This is one of the major limitations associated with web filters as identified in the section labeled 'Frequency of false negatives and Frequency of False positives', as shown in Figure 4.1; majority of the respondents raised the challenges of false positives (40.5%) and false negatives (46.2%). These results call for attention of the future web content filtering developers to ensure that newer filtering systems have improved filtering system.

4.3.4 Are web filters that important?

Answers to this question are as depicted in some sections of Figure 4.1:

i. **Impact of Web Filters** on employees' monitoring, the policies and rules enforced through Internet Filtering, the importance of content filtering and on the effect on data leakage prevention.

ii. **Productivity Enhancement**: This is concerned about whether Internet filtering enhances productivity and reduces distractions. The 'Productivity' section shows the effect of internet filtering on employee's productivity. Out of a total response of 158, 141(89.2%) responses agree that filtering out non-work related contents aids productivity in the working place.

4.3.5 User Security Awareness

Certain responses to the questionnaire were analysed to determine the effective control measures on social networks.

i. **Incident reporting** as reported in Figure 4.1 shows the frequency of reporting incidences by users. There is a low attention to activities reporting in instances of false positives and false negatives, and regrettably, 82% of the population are not aware of any reporting mechanisms, while only 43.3% takes incident reporting serious. The section labeled 'Incident Reporting' shows the frequency of reporting incidences by users.

ii. **Privacy Settings** which includes Password Security and 2FA' in Figure 4.1 is required to confirm the relevance of regular updated privacy settings on users' social media accounts,

and its importance in controlling access to certain information about the owner of the account. About 114 of 161 respondents representing 70.8% agree that the update of privacy settings is an effective measure to secure the social network space, confirming that privacy setting is an effective tool for personal information on social networks.

iii. **Password Security:** This is required to confirm the strengths of user's password in social media accounts. As presented in the same Figure 4.1, a total number of 107 representing 66.9% have very strong passwords. This is encouraging as it aligns with the International Data security body (Khando et al., 2021).

iv. **Two-Factor Authentication (2FA):** The associated questions were to test the percentage of users that enable two-factor authentication (2FA) to secure their social media accounts. A total of 151 respondents representing 93.8% subscribed to the use of 2FA to secure their social network against cyber threats and other security challenges.

4.4 Results from Model Development (Egenti-Filter Web Browser Extension)

Figure 4.2 presents the graphical user interface of the developed Egenti-Filter, which is an easy way to interact with the developed model so as to harness its diverse functionalities. The Graphical User Interface (GUI) has different sections including the CRUD operation, the database update, the database activity summary, the crawler, and the model performance statistics section.

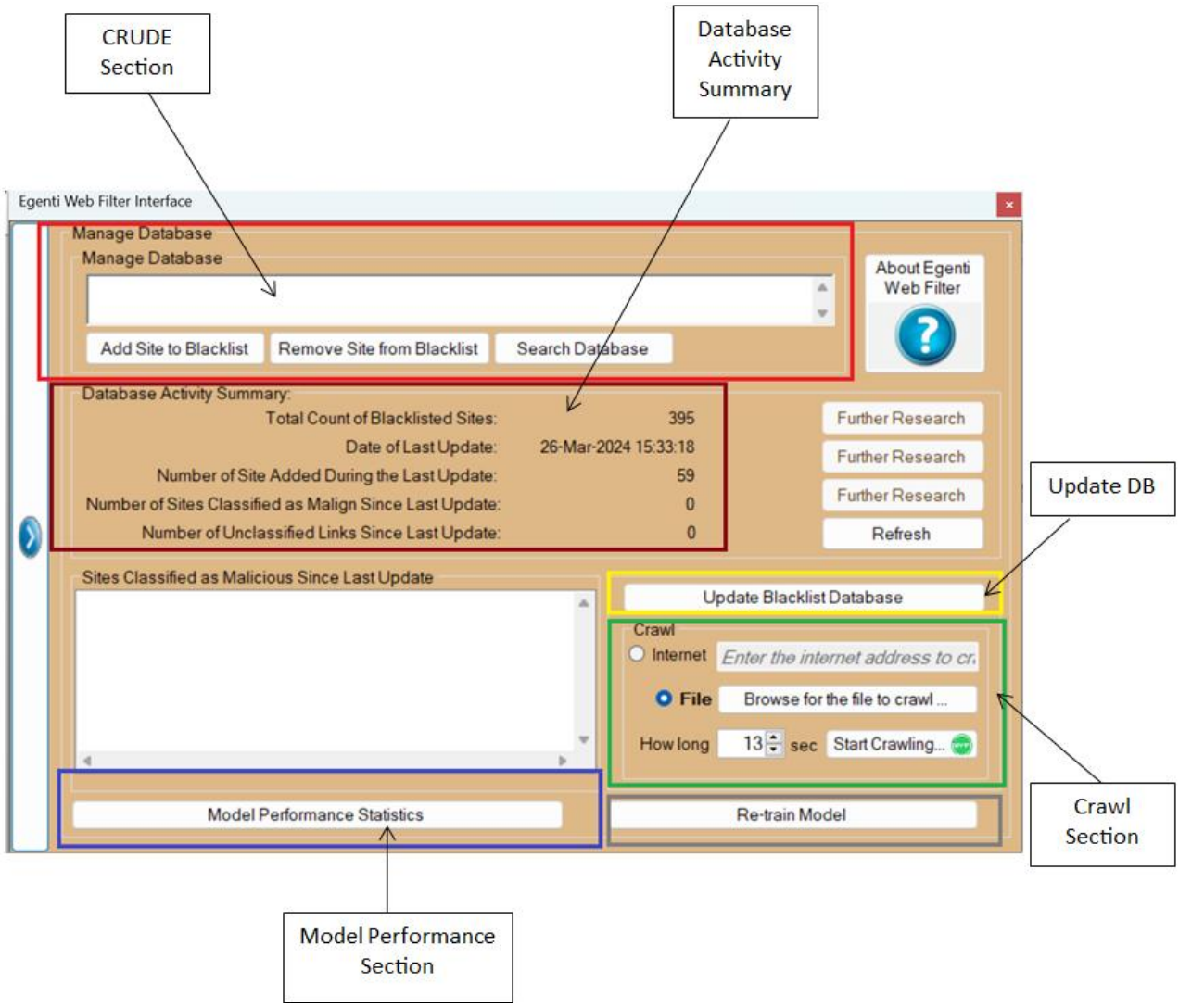


Figure 4.2: GUI Sections of Egenti-Filter

- **CRUD Operation Section:** Performs the Create, the Retrieve, the Update and the Delete operations, similarly carried out on all databases. Operations, such as the addition and removal of malicious links to and from the blacklisted database; searching for a specific website in the database, and updating the blacklisted database, are all features of the GUI.
- **Updating the Blacklisted Database:** As more websites and documents are crawled and classified, and more blacklisted URLs are found; there is the need to update the database regularly, and it is being populated with the latest content. Figure 4.3 shows that the database can be manually updated, and automatically updated by directly crawling websites, classifying the websites and the addition to the database.

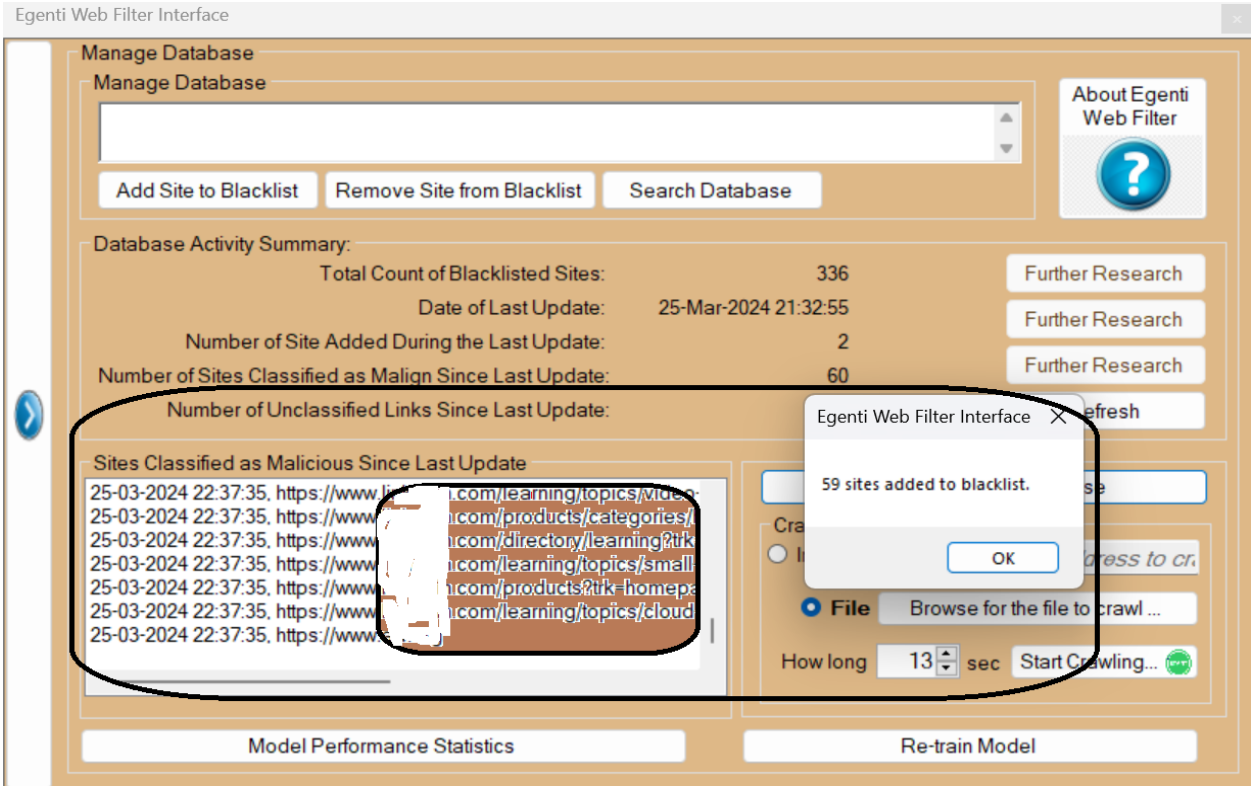


Figure 4.3: Updating of the blacklisted database

- **Database Activity Summary Section:** The Database Activity section is a very important aspect of this Interface; it displays the summary of the activities that had happened in the database over a period of time; for example, Figure 4.4 shows the total count of blacklisted URLs in the database at the moment, the timestamp when it was last updated and the number of links added.

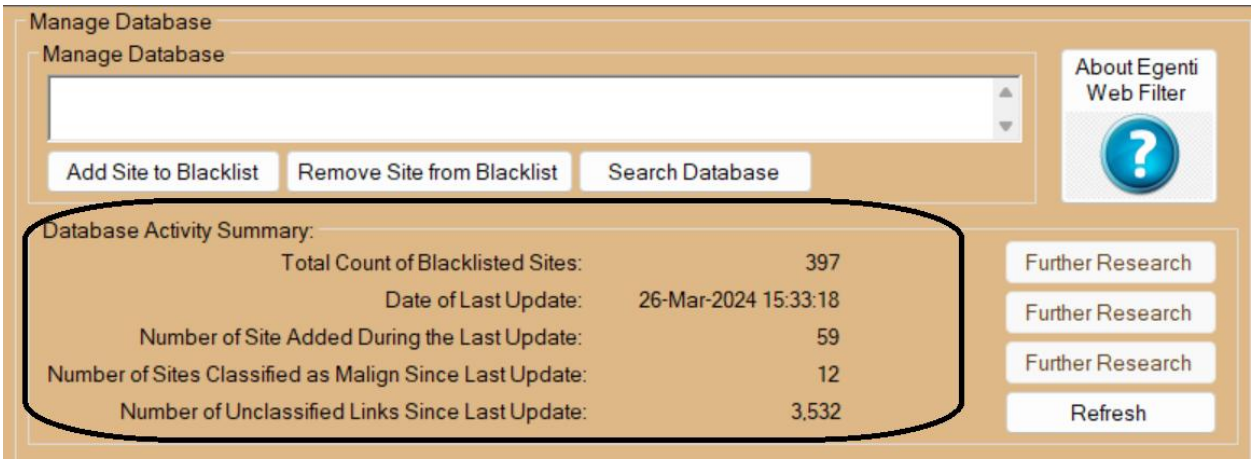


Figure 4.4: Database Activity Summary

- **The URL Crawler Section:** The Model is built to crawl a specific website on the Internet and/or a file or document saved on the user's local system, Figure 4.5 shows the crawling section of the Egenti-Filter Interface, with the option to either crawl the Internet (website) or a document. The default minimum crawl time is 5second; the speed of the crawling also depends on the Internet speed of the user. Figures 4.6 and 4.7 show the unclassified crawled links and the found malicious links after classification respectively. Figure 4.8 shows the document crawling option. For ethical reasons, the URL being crawled is blocked.

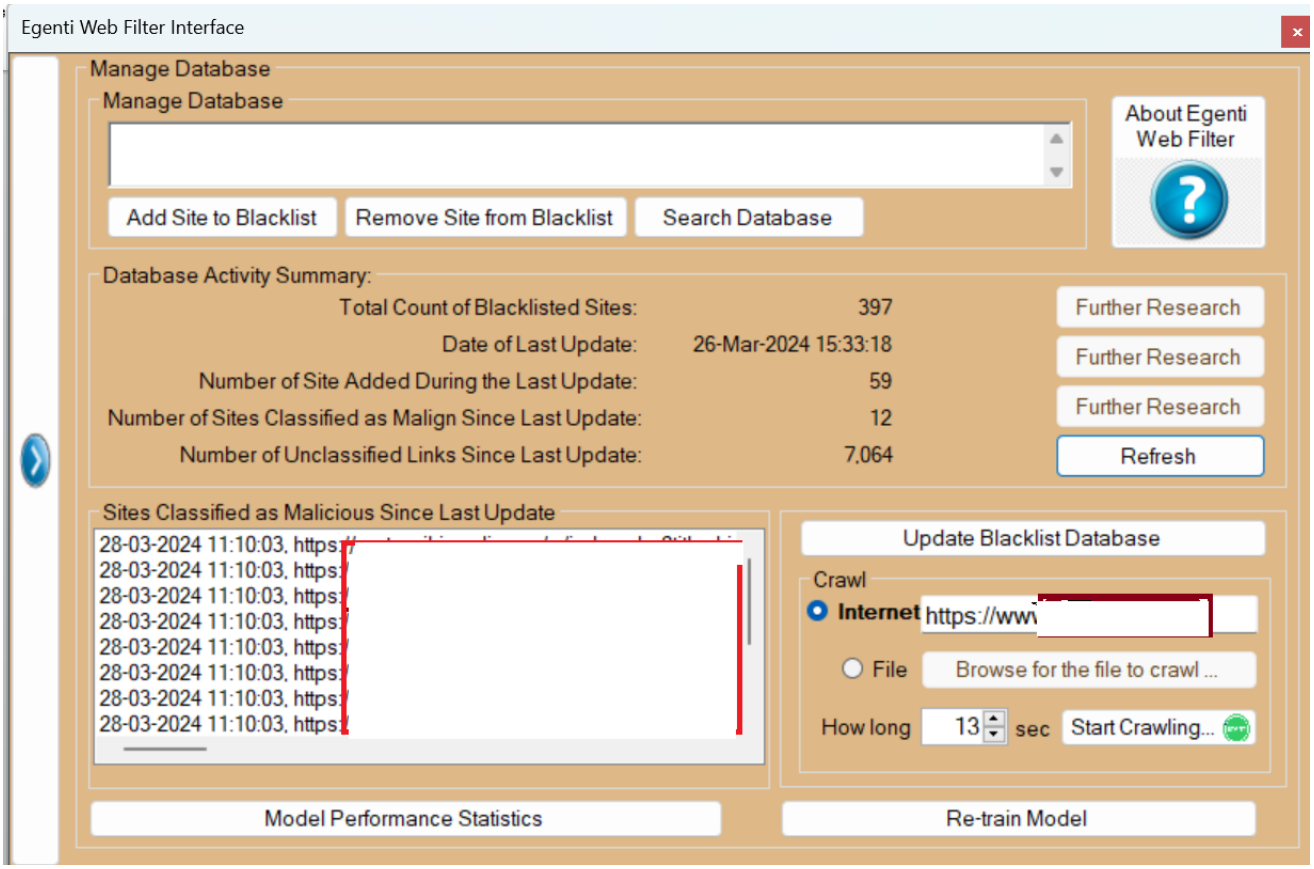


Figure 4.5: Website for Crawling

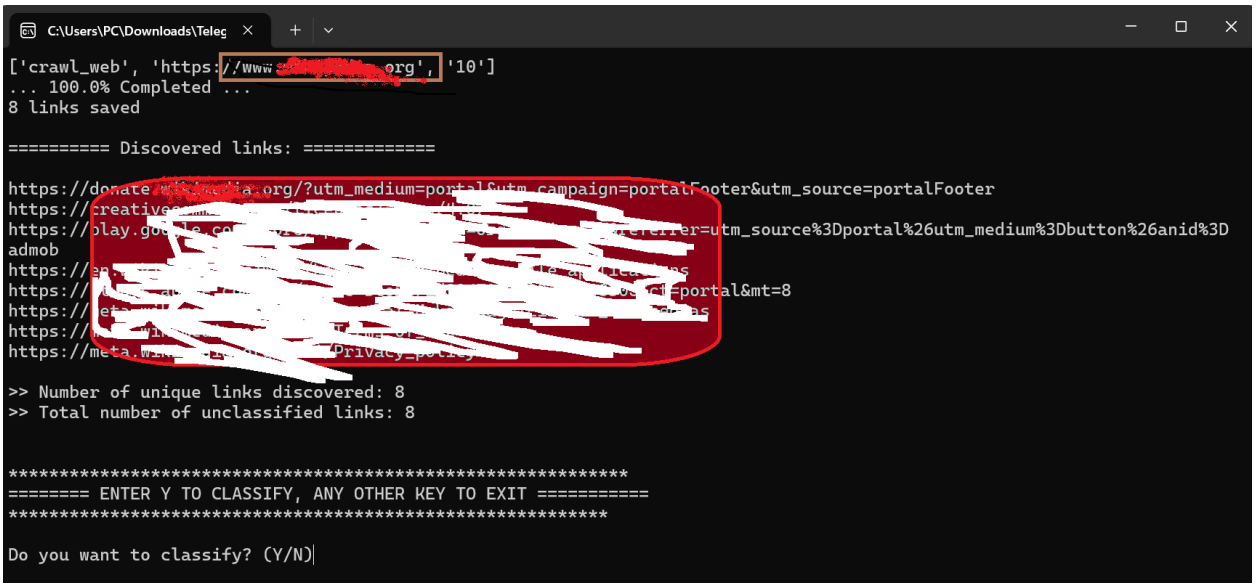


Figure 4.6: Unclassified Crawled links

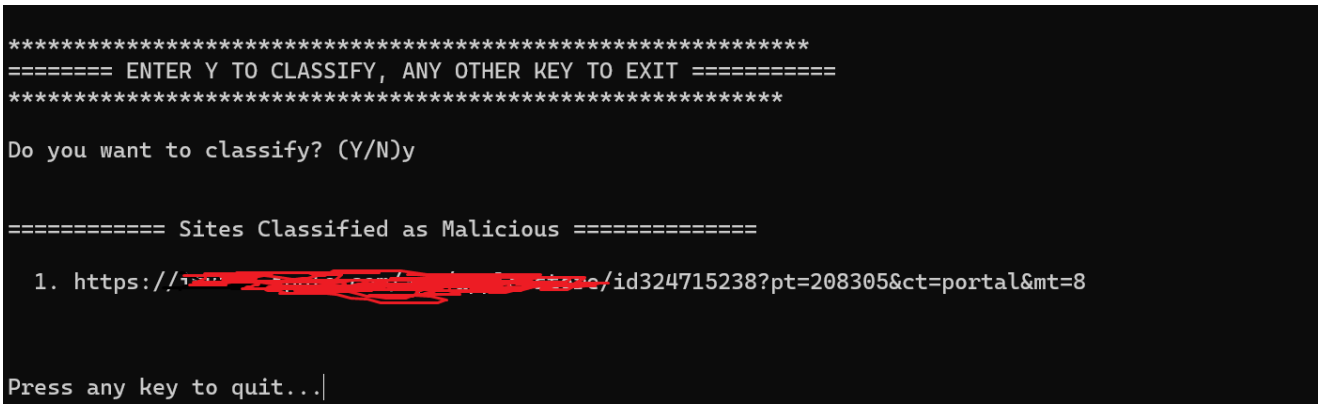


Figure 4.7: Displayed 1 malicious link after classification

- **Crawling a Document:** If the option is to crawl a document on the user's system, Figure 4.8 depicts the operation. As soon as the crawling is complete, the links are automatically added to the Dashboard, and the database is updated as needed.

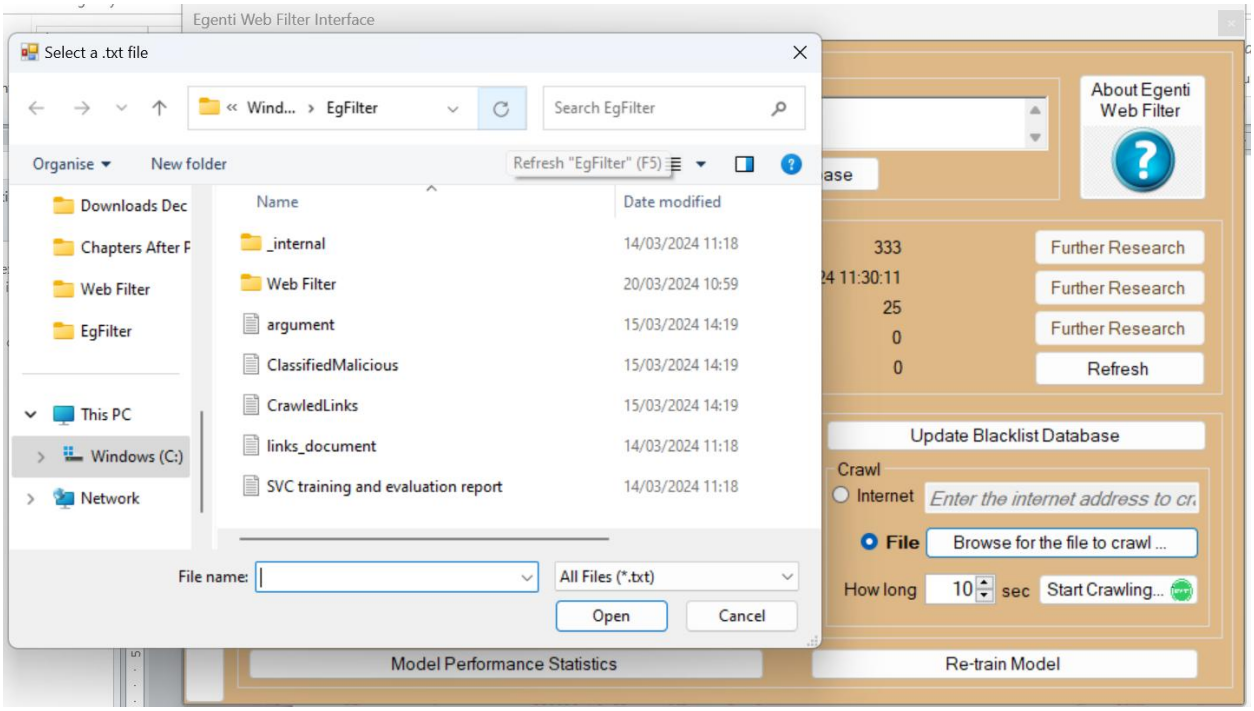


Figure 4.8: Document Crawling option

To affirm that the developed Egenti-Filter could protect online users from blacklisted sites, the author attempted to visit the listed links (Appendix E) in the database but without success as access was logically denied, because Egenti-Filter was installed on the browser used to visit the links. This denial resulted in the display of the restricted web page shown in Figure 4.9; this is the index file that is displayed each time a blacklisted website is visited by the user. This showed that the online user that has extension installed will not be able to access the blacklisted websites and is therefore protected from online threats (malware, data loss, data leaks, phishing and social engineering, browser hijacking, identity theft, etc.). Egenti-Filter tracks an online user's activity, and was programmed to block specific websites found in the database, and locking out an end-user from accessing the intended website, hence, protecting the individual, the organisation against lost productivity, network bandwidth issues, and possible legal issues that may arise as a result of the misuse of organisation's Internet resources. More so, Egenti-Filter offers tools to tailor any user's filtering preference to comply with the organisation's Internet usage policy. Therefore, online users can significantly limit the risk of exposure and falling victims to online security threats.

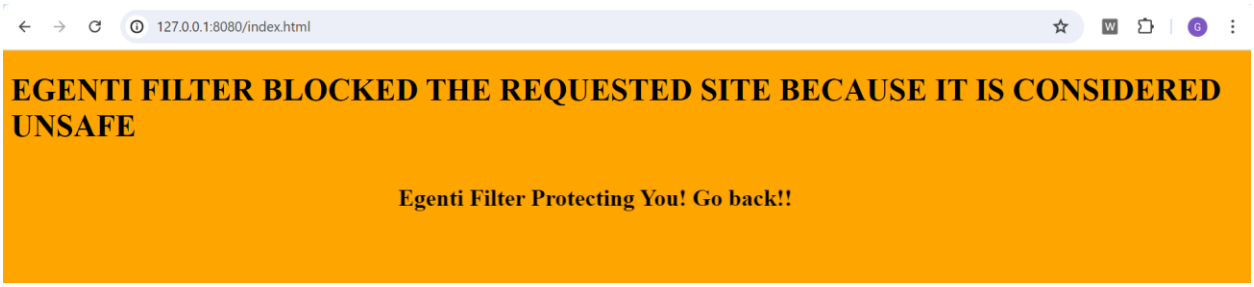


Figure 4.9: Restricted website by Egenti-Filter Browser Extension

4.5 Performance Evaluation of Egenti-Filter Model

The performance evaluation of Egenti-Filter used the classification performance metrics, such as accuracy, precision, recall and f1-score to provide insights into how well it performed in distinguishing between benign and malicious websites. These performance metrics are frequently used in machine learning to assess how well categorization models perform, and offer a methodical approach of evaluating the effectiveness of a model in accurately categorizing occurrences into the relevant groups. They provide supplementary data regarding a classification model's performance and are regularly combined to offer a more thorough assessment. When working with imbalanced datasets or when the cost of various error kinds (false positives, false negatives) varies, they are especially helpful. Precision focuses on avoiding false positives, while recall focuses on reducing or minimizing false negatives. The accuracy of the developed model is very paramount to have a sensitive robust model with high performance; however, accuracy is dependent on the quality of the dataset. The weighted average f1-score considers both precision and recall across classes. Overall, emphasis is placed on the learning rate and thorough evaluation to ensure that the model is performing at a high accuracy and is ready for deployment in real-world scenarios on edge devices.

This following section discusses the training and validation losses during the model development, the performance evaluation parameters used, and the result of the testing Egenti-Filter before and after model serialization.

4.5.1 Training and Validation Losses

The training regimen spanned one hundred meticulously calibrated epochs to achieve the optimal model performance; this indicates that the training process comprised putting the model through the training dataset 100 times, carefully adjusting each time to maximize the model's performance. This shows that during each iteration, the losses in Egenti-Filter was inversely

proportional to the iterations; which means that error that would have generated misleading results was sufficiently reduced, confirming that the model performed relatively well with the dataset. The model was validated with the validation dataset where 20% of the dataset were used for the testing and validation.

4.5.2 Egenti-Filter Evaluation Parameters

The Graphical User Interface of Egenti-Filter (Figure 4.2) has a section where the performance is evaluated. The Figure 4.10 provides an interface for evaluating the performance of the trained model, while Figures 4:11 and 4.12 show the snippets for computing performance metrics, and Figure 4.13 depicts the model parameters used for the evaluation and the results.

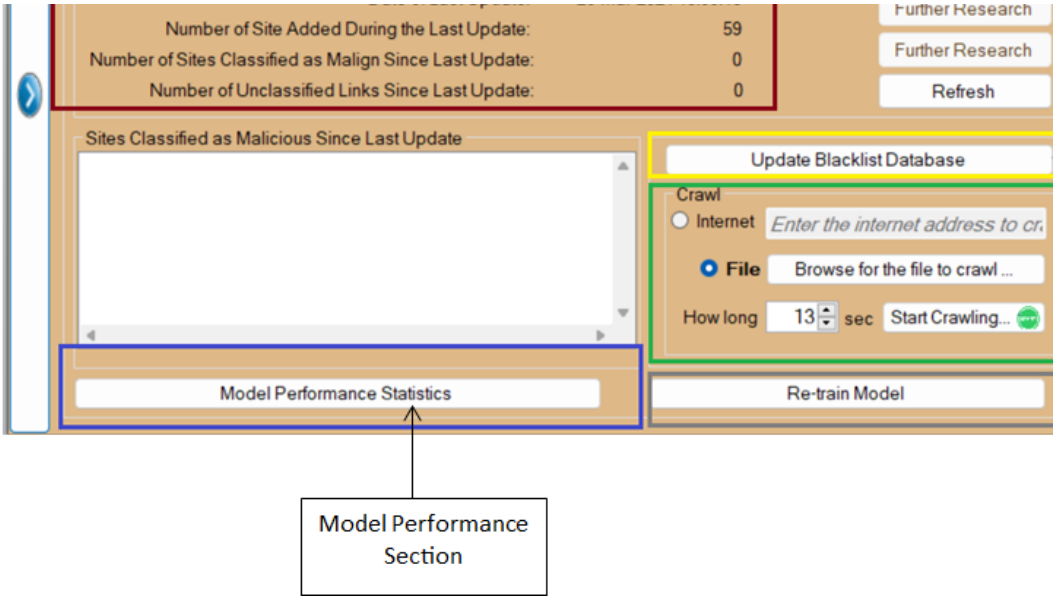


Figure 4:10: Egenti-Filter Evaluation Statistics

```
def compute_performance_metrics(TP, FN, FP, TN):
    # Total number of samples
    total_samples = TP + FN + FP + TN

    # Accuracy
    accuracy = (TP + TN) / total_samples

    # Informedness (Youden's J statistic)
    J = (TP / (TP + FN)) + (TN / (TN + FP)) - 1

    # Prevalence threshold
    PT = (TP + FN) / total_samples

    # True positive rate (Sensitivity, Recall)
    TPR = TP / (TP + FN)

    # False positive rate
    FPR = FP / (FP + TN)

    # False negative rate (Miss rate)
    FNR = FN / (FN + TP)

    # True negative rate (Specificity)
    TNR = TN / (TN + FP)

    # Negative likelihood ratio
    NLR = FNR / TNR if TNR != 0 else float('inf')
```

Figure 4.11: Snippets for computing performance metrics

```
return {
    'Total Population': total_samples,
    'Accuracy': accuracy,
    'Precision': precision,
    'Recall': recall,
    'F1-Score': f1_score,
    'Informedness': J,
    'Prevalence threshold': PT,
    'True positive rate': TPR,
    'False positive rate': FPR,
    'False negative rate': FNR,
    'True negative rate': TNR,
    'Negative likelihood ratio': NLR,
    'Positive likelihood ratio': PLR,
    'Markedness': MK,
    'Diagnostic odds ratio': DOR,
    'Matthews correlation coefficient': MCC,
    'Threat score': TS,
    'Prevalence': P,
    'Balanced accuracy': BA,
    'Positive predictive value': PPV,
    'False discovery rate': FDR,
    'False omission rate': FOR,
    'Negative predictive value': NPV,
```

Figure 4.12: More snippets for computing performance metrics

```
============ model evaluation results ==============

[[2143   80]
 [ 525  994]]

              precision    recall  f1-score   support

      benign       0.80      0.96      0.88      2223
   malicious       0.93      0.65      0.77      1519

    accuracy                           0.84      3742
   macro avg       0.86      0.81      0.82      3742
weighted avg       0.85      0.84      0.83      3742
```

Figure 4:13: Model Performance Evaluation Results

The Precision for the benign class (non-malicious) is approximately 80%, meaning that out of all predicted benign instances, 80% are actually benign. Recall for the benign classification is approximately 96%, indicating that Egenti-Filter correctly identified 96% of actual benign instances. The f1-score for the benign class is approximately 88%, which balances precision and recall. For the malicious class, precision is approximately 93%, recall is approximately 65%, and the f1-score is approximately 77%. - The overall accuracy of Egenti-Filter is approximately 84%. Figure 4:14 shows more results from the model performance evaluation.

```
============ Model performance statistics ==============

TP: 2143, FN: 80, FP: 525, TN: 994
                          Total Population:  3742.0000
                                  Accuracy:     0.8383
                                 Precision:     0.8032
                                    Recall:     0.9640
                                  F1-Score:     0.8763
                       Prevalence threshold:     0.5941
                         True positive rate:     0.9640
                        False positive rate:     0.3456
                        False negative rate:     0.0360
                         True negative rate:     0.6544
                  Negative likelihood ratio:     0.0550
                  Positive likelihood ratio:     2.7892
                                 Markedness:     0.6184
                      Diagnostic odds ratio:    50.7177
          Matthews correlation coefficient:     0.6713
                              Threat score:     0.7798
                                Prevalence:     0.7130
                        Balanced accuracy:     0.8092
                  Positive predictive value:     0.8032
                       False discovery rate:     0.1968
                        False omission rate:     0.0745
                  Negative predictive value:     0.9255
                      Fowlkes-Mallows index:     0.0190
```

Figure 4:14: More Evaluation Metrics

The model performance statistics offer significant insights into the model's classification performance, as shown in Figure 4.14:

i. Accuracy: The model's overall accuracy is roughly 83.83%. The percentage of correctly identified cases in the overall population is represented by this metric.

ii. Precision: About 80.32% is the precision for the benign class (non-malicious). This indicates that 80.32% of all cases that were expected to be benign actually are.

iii. Recall (Sensitivity): About 96.40% of the benign class' recall is available. It shows that 96.40% of true benign cases are appropriately identified by the model.

iv. F1-Score: About 87.63% is the F1-score for the benign class. This measure offers a single number to evaluate overall performance by striking a balance between recall and precision.

v. Balanced Accuracy: About 80.92% is the balanced accuracy. It provides a balanced perspective across all the links by taking into account both real positive rate (recall) and true negative rate.

vi. Correlation Coefficient of Matthews (MCC): 0.6713 is the approximate MCC. Giving a thorough assessment of categorization performance of the model; it considers true positives, true negatives, false positives, and false negatives.

4.5.2b. The significance of statistical findings on real-world applications

Accuracy (83.83%) - Total Performance of the Model: Eighty-three percent of all URLs, both benign and malicious are accurately classified by the model. A high accuracy indicates that the Egenti-Filter is effective against a variety of threats. An accuracy of 83.83% in real-world applications indicates that, on average, 8 out of 10 websites that are visited by users are appropriately categorised as either safe or malicious. This helps stop drive-by downloads, phishing schemes, and other online threats.

The precision (80.32%) of the Egenti-Filter in identifying malicious websites: About eight out of ten threats that are flagged are successfully identified. High precision is essential for user confidence. Users may disable the model if it incorrectly blocks an excessive number of safe URLs. Sensitivity to Benign Websites and Recall (96.40% for Benign Class): The model's recall, sometimes referred to as sensitivity, gauges how well it can identify safe websites without mistakenly labeling them as dangerous. Nearly all safe URLs are correctly categorised, lowering false alerts, with a recall of 96.40% for benign websites. It makes the benign URLs available, thereby minimising interference for authorised users. Also maintaining seamless online browsing without frequent disruptions requires high recall. A lower recall, however, would result in the needless blocking of numerous safe URLs, which would negatively impact user experience.

F1-Score (87.63%): Finding a Balance between Recall and Precision which is a harmonic mean of the two. The model's 87.63% F1-score indicates that it successfully reduces false positives and false negatives, guaranteeing high overall reliability. The model is well-optimized if the F1-score is high, which lowers the number of missed threats (false negatives) and needless blocks (false positives). In high-stakes situations, this helps cybersecurity teams have faith in the system's filtering skills.

4.5.3 Testing Egenti-Filter before and after Serialization

Figure 4.15 shows the result of testing model before serialization and testing model deployment after serialization. Serialization has to do with the conversion of the trained Egenti-Filter into a format that can be stored or transmitted and reconstructed later for deployment, the model's parameters, architecture, and other essential components are saved to disk or memory. Testing the model before serialization is to ensure that it is working correctly as expected and accurate predictions are made on the dataset. Testing the model after serialization ensures that the serialized model can be loaded and used correctly in the deployed environment; this is to maintain the model's performance and reliability in production environment. In essence, Figure 4.15 depicts the importance of testing machine learning models at different stages of the development and deployment process, both before and after serialization, to ensure the model's consistency and reliability, thus enhancing model performance.

```
============ Testing Model Before Serialization ==============
https://telewizjada.net : benign
 https://btc4kids.ru : malicious
   https://95095.com : benign
 https://nhentai.net : benign


============ Testing Model Deployment After Serialization ==============

https://telewizjada.net : benign
 https://btc4kids.ru : malicious
   https://95095.com : benign
 https://nhentai.net : benign
```

Figure 4.15: Testing model before and after serialization

4.6 Summary of the Features of Egenti-Filter

The features of Egenti-Filter browser extension are summarized:

i. URL Classification: Detects and classifies URLs as either malicious or safe based on predefined rules. It utilizes a rule-based system to match URLs against blacklists and whitelists for immediate classification.

ii. Dynamic Rule Generation: It automatically generates and updates filtering rules based on user feedback, model predictions, and threat intelligence data. It adapts to evolving threats and user preferences by learning from new URLs and user interactions.

iii. Real-time Blocking: It blocks access to malicious URLs in real-time, preventing users from visiting potentially harmful websites. It displays warning pages with risk indicators and safety recommendations for flagged URLs.

iv. User Feedback Mechanism: It allows users to report URLs as safe or malicious, providing feedback to improve classification accuracy and rule effectiveness. It incorporates user feedback into the training pipeline for model retraining and rule refinement.

v. Configuration Options: It provides customizable settings for users to configure filtering preferences, manage blacklists/whitelists, and adjust filter sensitivity. It enables users to fine-tune filter behaviour, update rule databases, and view classification history.

vi. Help and Documentation: It offers help resources, documentation, and tooltips within the extension interface to guide users on using the filter effectively. It provides educational content on web security, URL classification, and safe browsing practices.

vii. Performance Metrics: It measures and displays performance indices, like accuracy, precision, recall, and f1-score to evaluate filter effectiveness. It enables users to assess the reliability of classification results and monitor filter performance over time.

viii. Browser Compatibility: Egenti-Filter is compatible with major web browsers (Chrome, Firefox, Microsoft Edge, etc.,) as a browser extension, ensuring broad accessibility and usability for users.

ix. Automatic Updates: It automatically updates rule databases, machine learning models, and filter configurations to stay up-to-date with emerging threats and security trends. It enhances security posture by incorporating the latest threat intelligence and model improvements.

x. Offline Mode: It provides offline mode functionality, allowing the extension to operate without internet connectivity by using locally stored rule databases and cached model predictions. It also ensures continuous protection and URL classification even in offline environments or low-connectivity scenarios.

CHAPTER FIVE

DISCUSSION OF RESULTS

5.1 Introduction

This chapter presents the discussion based on the outcome of this research endeavour in terms of benchmarking and comparing Egenti-Filter with our similar studies. Answers to research Questions were also discussed. Practical contribution to knowledge and the study limitations were all discussed.

5.2 Benchmarking of Egenti-Filter Results

The performance of the Egenti-Filter was compared with the results of Mathews and Chimalakonda, (2021), Norma, (2021), and Xuan and Nguyen, (2020). These three studies were chosen because the aspects of a wider challenge that is centred on online safety were represented; while the approaches differ slightly, the overarching objective of enhancing online safety is covered by all three studies. For instance, the primary focus of the first study (Mathews and Chimalakonda, (2021), and the third study (Xuan & Nguyen, 2020) is centred on malicious website blocking, which is also the focus of this research work. The second study (Norma, (2021)) focuses on content filtering that has the propensity to negatively impact mental well-being of individuals. This is also an aspect of this research work. In terms of the technologies used, all three studies leverage machine learning algorithms to varying degrees of usage, this research work also leverage machine learning for the model development. In essence, all three studies have close to the same aim of this research endeavour.

The aim of the work conducted by Mathews and Chimalakonda, (2021) was to improve people's mental health by proposing the ability to take control of the content available to viewers on the Internet. This aim is similar with our work which also focused on the ability of users to take control of what is exposed to them on the Internet. The identified limitations are listed:

i. The experiment did not perform as expected, as it caused lots of false positive results.
ii. To resolve challenge in (i), the researchers analyzed the whole content, this approach tends to be more resource-intensive, and so the approach was abandoned.
iii. The categorizer was trained on Indian headlines; as a result, the experiment is region-specific.

iv. This experiment works only with website in English language content but does not support local languages.
v. The tool used could be adopted by the public, but the researchers could not comment on the efficacy of the experiment.
vi. There is need to identify features for the websites to be analyzed and added to the database as appropriately required, meaning that the addition of new websites was done with manual effort.
vii. Major updates on how the content is displayed on the target website may distort the primary scripts.
viii. To resolve the challenge in (vii), the researchers have to keep checks on websites that display warning messages when the patterns no longer return the expected results.

In comparison with this study:

i. The Egenti-Filter was developed and implemented and was not abandoned.
ii. Egenti-Filter is a lightweight Javascript which is less resource intensive.
iii. Egenti-Filter is not language specific, as it can filter non-English websites, and can be deployed everywhere without being region-specific.
iv. The model worked well when it was tested with 20 malicious websites.
v. As against manual updates performed on the database, Egenti-Filter performs both manual and automatic updates.

To address the issue of traditional approaches not been able to frequently update the blacklisted database, this researcher, Norma, (2021) proposed a system that could be plugged into the Google Chrome extension or Firewall. The researcher began by using dataset made up of malicious and benign websites; features were extracted from these websites which were analyzed and pre-processed to generate easily separable and classified data. Results from this experiment revealed a detection rate of 99.88% of malicious links and false hit rate of about 2.61%. Again, in line with Norma, (2021), Chiba et al., (2012), proposed a system that could detect malicious or benign websites only by analyzing IP features/characteristics. The researcher created a dataset that extracted only campus traffic, and pre-processed the data by separating the IP addressed into bits and applied two different Machine Learning algorithms (the Support Vector Machine and Restricted Boltzmann Machines). It was recorded that a maximum of 90% accuracy was achieved. Although, Egenti-Filter used the support vector machine for website classification, but

IP addresses were not used, rather URLs was used which gave more features and produced a more precise model than that proposed by Chiba et al., (2012), thus:

i. Only the IP characteristics were considered
ii. Only campus traffic dataset was used for the experiment
iii. 90% accuracy was achieved

In the study conducted by Xuan and Nguyen, (2020), machine learning algorithms were employed to categorise URLs based on the features and behaviours of such websites, and for the detection of malicious websites. It was also recorded that Support vector machine (SVM) and Random forest (RF) were the two supervised machine learning algorithms that were used for the experiment to extract dynamic behaviours of the URLs. Similarly, the SVM and the RF algorithms were also used for the detection and classification of websites. The contrast between this and Egenti-Filter is that the researchers used far more URL features than was used by Egenti-Filter, which also increased the computational resources of the network. Interestingly, Xuan and Nguyen, (2020), equally obtained high performance accuracy rate of 93%, having a 4% less recall rate.

i. Far more URL features were used for the experiment
ii. Increase in computational resources of the network
iii. High performance accuracy rate of 93%
iv. A 4% less recall rate from Egenti-Filter.

Egenti-Filter was also compared with the study conducted by Thangaraj, (2019) too, the study presents a novel web content filtering algorithm that classifies pages as either approved or forbidden, protecting users from the possible risks associated with uncontrolled internet access; based on the filtering criteria, the 'KT–Grand' algorithm evaluates the content of web pages and determines whether to provide or deny access to the page. The goal of the two-pronged strategy of email and web filtering is to make the internet safer for users. Appendix F summarises the performance of related works in comparison with similar study.

Table 5.1: Comparison of Egenti-Filter with an Early Decision Algorithm

Attributes	An Early Decision Algorithm Thangaraj, (2019)	Egenti-Filtering Model(This Project)
Dataset Source	Web contents are taken from Directories (yahoo.com)	Web contents are taken from URL in real time (Online and offline)
Filtering Process	The filtering process scanned only the front part of the Web content	The filtering process scanned the entire part of the Web content using Web crawler
Search Process	It searches the desired keywords only	It searches the desired keywords and also its corresponding synonym words
HTML Words	This Algorithm ignores all HTML words	HTML words are considered because unwanted illegal content may be inserted in the web pages HTML tags by the creators.
Site Classification	Does not classify non-English Websites	Classifies Non-English websites
Flexibility/ User Friendliness	Early Decision algorithm is less flexible because it is a pre-defined filtering algorithm	The Browser Extension filtering method is more flexible and customizable because it can be customised to individual preferences.
Update	Early Decision algorithm would require more substantial changes to the algorithm itself, which may be more difficult and time-consuming.	Egenti-Filter is easily updated and adapted to reflect the changes than the Early Decision algorithm.
Granularity	Filtering is done on pre-defined criteria.	Provides more granular and nuanced filtering options than the Early Decision algorithm.
Transparency	Early Decision algorithm may be more opaque and difficult for users to understand or modify	More transparent and user-friendly than the Early Decision algorithm.

Google has an inbuilt feature, commonly referred to as 'safe search'; this feature expunges search results that are considered age-inappropriate for users. Although, safe search feature sieves out undesirable content, unfortunately, it does not allow for customisation based on users' preferences. Also considering the massive online content that has to be filtered out, Google's safe search is not sufficient enough to handle this. Hence, to extend its functionality, certain chrome extensions (Profanity Filter, Safe Words, etc.,) was used to improve the feature of safe search, which can remove profane words, and perform filtering with symbolic stand-ins to obfuscate these words (Athirah & Suliman, 2017). These extensions focus on removing profane words but not on other types of content that the user might want to filter or might be sensitive to. Other tools such as the Good News Chrome extension5 which blocks news stories based on keywords and phrases, but this works only on the Google News website. Further, to the best of the author's knowledge, there has not been much effort towards the use of browser extensions for blocking entire website content that works well on the returned search results that enable users to have better control over their web browsing experience; hence, the usefulness of Egenti-Filter.

5.3 Answers to the Research Questions (RQs)

This section presents different findings and answers to this study's Research Questions. The recap of the research questions are stated in section 5.3.1, while the different findings and answers are presented in section 5.4. An overview of the research questions and answers are presented at a glance in Table 5.2 of section 5.5.

5.4. Recap of Research Questions

The primary research question for this study as introduced in section 1.3 is:

How can web filters be used to effectively secure online activities of social networkers?

Consequently, the under-listed secondary research questions were identified to unravel the challenges surrounding online activities and proffer answers to the primary research question:

i. Which current security control measures are effective for securing online activities?
ii. Are web filters effectively securing online activities?
iii. What are the major limitations of web content filters?
iv. Why are web content filters considered the best security approach for online users?

5.5. Response to Research Sub-questions

This section focuses on considering the main findings since it is linked to the individual research question towards providing most appropriate response to each of the study questions.

5.5.1

Which current security control measures are effective for securing online activities?

This question was tackled in two ways – by conducting a comprehensive literature review; as well as a survey among the cybersecurity professionals. Various literatures (Alzubaidi, 2021, Faruk et al., 2021; Herath et al., 2022; Nicolaidou & Venizelou, 2020) show substantial vulnerabilities amongst online users and emphasize cyber insecurity as a pressing issue in social networking. It also confirmed that attackers are continually developing new strategies, and that various control measures (passwords, PIN, privacy settings, 2FA, etc.,) are now in place to mitigate cyber-attacks. These measures are necessary to guard against known threats such as attacks via malware or viruses, direct network intrusion, or illegal monitoring of unsecured communications.

Many scholars (Etuh et al., 2021; Kashmar et al., 2022; Ogundele et al., 2020; Okesola, 2014; Vivekanandam & Midhunchakkaravarthy, 2022) have proved that the existing online security measures are not adequate to fully protect users and data. However, the survey analysis amongst information security professionals show a significant relationship between online security and web content filtering, thereby confirming the web content filtering approach as the most effective cybersecurity tool of preventing cyber-attacks in personal computers, organisations' network, or school library systems (Section 2.6). Other security approaches - credentials authentication (username, password, PIN), Role Based Access Controls (RBAC), Risk-based access control, etc., are also adjudged effective but with some drawbacks that are counterproductive to the security objectives.

While different innovative approaches to cybersecurity, such as spam detection, cyberbullying detection, phishing detection, and cyber grooming recognition have been developed in the academia, they often lack practicality for real-world application (section 2.12). Although, strong security technologies are also considered effective, they are insufficient to prevent security breaches (Okesola, 2014) because of the yearly continuous growth in cyber incidents,

exploitation of common security weaknesses, evolving attack methods, insider threats, and limitations of individual controls.

5.5.2

Are web filters effectively securing online activities?

Yes, Web filters have been confirmed effective to secure online activities (Figure 4.9). The performance evaluation of Egenti-Filter model, developed in this study, adopted the classification performance metrics of Precision, Accuracy, f1-score and Recall to provide insights into how well the model could perform in distinguishing benign from malicious websites. The Performance evaluation recorded the following achievements:

i. Accuracy - the model's overall accuracy was 83.83%.
ii. Precision - about 80.32% is the precision for the benign class (non-malicious).
iii. Recall (Sensitivity) - about 96.40% of the benign class' recall is available.
iv. F1-Score - about 87.63% is given as the f1-score for the benign classification. This measure offers a single number to evaluate the overall performance of the model by reaching a compromise between the recall and precision.

Attempted visits to 20 blacklisted links in the database were promptly denied by Egenti-Filter Whereas, visits to the same 20 blacklisted websites where the Web browser extension was not installed were successful with ease thereby making the online user vulnerable to online threats and attacks. The survey analysis as presented in Figure 4.1 shows an improved employee productivity when web filtering is installed, by preventing employees from visiting unrelated websites during office hours, reduced data leakage, reduced exposure to cyber risks, and malware infection.

5.5.3

What are the major limitations of web content filters?

Some past related studies (Ali et al., 2021; Diaz-Garcia et al., 2022; Overhaul, 2022, etc.) and survey conducted in section 3.3(i) have achieved appreciable results in online security, although with some associated challenges that induced researchers to propose more effective security

mechanisms like the multi-factor authentication (MFA), and web content filters (Petru-Cristian, 2023; Zwilling et al., 2022). However, as effective as Web content filter is to online access control and security, it is noted with some shortcomings that may deter its usability and general acceptability.

This study agrees with some past authors (Diaz-Garcia et al., 2022; Nicolaidou & Venizelou, 2020; Turner, 2021) that underblocking or over blocking of essential resources is a serious challenge to content filters. Tech-savvies can dwell on the dynamic nature of the internet to bypass many content filters. Hence, filtering regulations must be updated frequently to remain current with new dangers and evolving online habits. Meanwhile, manual database update is also a major challenge in existing web filters as it is cumbersome, error prone, and time-consuming, but now being addressed by our newly developed Web Filter -Egenti-Filter. Override permission presents additional difficulties as well since users might figure out how to by-pass web filters. Same is mediocrity in technology use for certain categories of users just as in parents with very low awareness of web content filtering. Even though, these limitations call for a more detailed approach to online safety, web content filtering approach remains crucial to Internet access controls and security.

5.5.4 Why are web content filters considered the best security approach for online users?

Past related studies (Chrysomalidis et al., 2021 & Deotte et al., 2021) have regarded content filtering approach as one of the best security solutions for online users. This position has been further established by Egenti-Filter for minimal false positives and false negatives towards shielding users from harmful websites. Considering the dynamic nature of the Internet, Egenti-Filter reacts swiftly to emerging risks by shifting users' behaviour, and strengthening defences against cyber-attacks, data breaches, and other security challenges. By avoiding distracting or non-work-related websites, the content filter also ensure adherence to legal and ethical standards towards fostering a productive online workplace.

Web content filters are significantly recognized as a better option to traditional online security solutions (antivirus firewalls, etc.,) especially in effectively managing and mitigating cyber-threats. Web content filters explicitly target and manage user access to web resources which allows organisations and individuals to block malicious or inappropriate websites, thus providing

a proactive defence against cyber-attacks. The superiority of web content filters over other solutions is postulated on Table 5.2.

Table 5.2: Comparison of Web content filtering with other online security solutions

Features/Characteristics	Firewalls	Antivirus	**Web Content Filtering**
Content Filtering	Only at the network level	No	**Yes**
Real-time threat protection	Moderate	Yes	**Yes**
Malware Prevention	Yes	Yes	**Yes**
User Management	Yes	Limited	**Yes**
Ease of setup	Moderate to complex	Easy	**Easy**
Phishing Detection	Yes	No	**Yes**
Network deployment	Yes	Limited	**Yes**
User Access Control	No	No	**Yes**
Content Classification	No	Limited	**Yes**
Cost-effectiveness	Higher for Premium features	Varies widely	**Often lower**
Customisation	Needs technical expertise	No	**Easy**
Granularity of Control	Low (allow/block lists)	Moderate	**High (provide more granular and nuanced filtering options)**

Our newly developed web content filter - Egenti-Filter - also possesses the following unique attributes amongst others:

i. With Egenti-Filter's intuitive Graphical User Interface (GUI), users can effortlessly modify their filtering options and get a simplified control experience. This interface makes it easier for users to customise filtering profiles in accordance with personal preferences or internet usage guidelines. The web content filter guarantees that users may efficiently navigate and control their online experiences while adhering to set guidelines by offering this degree of personalisation. This flexibility raises user satisfaction while

simultaneously strengthening the filtering system's overall efficacy in preventing the exposure of offensive contents.

ii. The online crawler and URL classifier of the Egenti-Filter are engineered to scan and categorise webpages in English and other languages thereby augmenting its capacity to offer all-inclusive filtering solutions. This feature guarantees users' access to a wide variety of content while upholding security across various languages and geographical areas. Egenti-Filter enhances its relevance and filtering accuracy by efficiently classifying websites in any language, enabling customised protection against offensive or malicious websites. The addition of multilingual support to the filtering system not only expands its functionality but also improves its flexibility in responding to different user requirements and organisational norms, making the filtering system a strong option for protecting online activities.

5.6. Summary Overview

The overview of the research questions and answers are presented in this section and highlights each research question alongside the respective research response.

RQ1: Which current security control measures are effective for securing online activities?
Answer**:** Passwords, PIN, usernames, 2FA, and Web browser extension

RQ2: Are web filters effectively securing online activities?
Answer: Yes; as represented in Figure 4.10, where access to 20 malicious websites were restricted after installing the Egenti-Filter, thereby securing the users from online exploitation, data breaches, malware infection, etc.,

RQ3: What are the major limitations of web content filters?
Answer: High rates of over-blocking (false positives) and under-blocking (false negatives); by-passing attempts; manual database updates; overriding permission and mediocrity in technology use for certain categories of users.

RQ4: Why are web content filters considered the best security approach for online users?
Answer: Minimal false positives and false negatives, swift reaction to emerging risks, strengthening defenses against cyber-attacks, adherence to legal and ethical standards, inclusion

of manual and automatic database updates, Egenti-Filter extension has a Graphical User Interface for easy control, modification and a tailored user's filtering profile, the web crawler and the URL classifier of Egenti-Filter is all-inclusive filtering, crawling and classifying both English and non-English websites.

5.7 Answer to the Primary Research Question

How can web filters be used to effectively secure online activities of social networkers?

The primary question for this study can now be handled since answers to the four sub-questions have been provided.

In furtherance to the challenges identified from the review of related literature in chapter two and the limitations of web content filtering approaches, a real-time web filtering model; Egenti-Filter was developed to mitigate the challenges, since it was developed as a proactive measure to secure the activities of online users using the listed components:

i. Inventing a Web Crawler for Link Extraction
ii. The URL Classifier
iii. The Machine Learning Module
iv. The Database Management Module
v. The Designing of the Filter's Graphical User Interface (GUI) and
vi. The Framework for Web Browser Extension (Egenti-Filter)

The conceptual model that relates these components influences the effectiveness of the security of online activities using the browser extension approach in this research. Different independent variables to the web browser extension worked harmoniously together to achieve the main objective of this research.

The designed research framework for the encapsulated Egenti-Filter captures the stages needed to develop a secure online activity using web content filtering approach (Figure 3.8). These phases were carefully crafted and implemented, and the resultant outcome was the development of a real-time web browser extension model for securing the online activities of users. For effectiveness however, this model has to be installed on the client's browser as an extension in order to harness the features of Egenti-Filter (browser extension).

5.8 Practical Contribution to Knowledge

The main contribution of this research to the domain of security and control measures is in the design and implementation of a unique web content filter using the browser extension approach. These contributions depict a clear-cut distinction and improvement from the existing web content filtering applications, and are categorized as primary and secondary.

5.8.1 Primary Contributions

The primary contributions of this study refer to the practical contribution that deals with the aim and objectives of this study; they are:

i. This study introduced a proactive measure of securing the activities of online users – a preventive approach where security of websites is determined before being accessed. The web content filter was developed, implemented, and verified effective to deny access to malicious websites.

ii. A web crawler was designed with genuine motive to crawl specific websites on the Internet. The choice for developing a web crawler for this study was necessary as the existing web crawlers are mostly made for general purposes, and data collected may not suite specified research goals. The custom crawler was built to capture only relevant information following the specified criteria such as the crawling depth which specifies the extent to which the crawler should delve into the link structure of the crawled website. More so, developing a custom web crawler makes it possible to test out novel web crawling algorithms and techniques. This may give rise to novel techniques that could enhance existing ones and an addition to the corpus of scholarly work in the domain of online security.

iii. The URL classifier was designed to categorize the URLs into malicious or benign links, which were used to update the database. The web crawler and the classifier systems may be made available to other researchers interested in this same area of study to monitor changes on websites over time by regularly revisiting the website to track any changes in the content. It may also be used as a digital archive of the web pages, thereby preserving information that may have changed, moved or deleted overtime.

iv. The model performance statistics offer significant insights into the model's classification performance, for instance, the accuracy was recorded as 83.83%, the precision (80.32%), the recall (96.40%) and the f1-score as 87.63%. These statistics may guide other

researchers to benchmark their works and make informed decisions towards improving their research quality.

5.8.2 Secondary Contributions

In carrying out this study, some probable challenges were discovered and called for immediate attention. The following solutions are categorized as secondary additions to the scientific body of knowledge, and are considered so significant:

i. A questionnaire for both existing and future use was produced (appendix C), which has focus on individuals who are experienced in the IT field, especially on cybersecurity and related fields. This questionnaire could be useful for researchers in a similar domain to determine the effective control measures on social networks, know the common web filtering solutions, ascertain if web filters are really that important, and measure the effectiveness of existing web content filters.

ii. This research would open up opportunities for collaboration as it may avail other researchers or developers working on related areas to interact especially where there are diverse perspectives of ideas, and gather combined efforts to deal with content filtering and web safety in general.

iii. Sharing Egenti-Filter as an open source project may inspire community engagement and additional improvements (such as new features, and fixing of bugs), and this may create additional documentation or guides towards promoting usability of this browser extension among new users. Comments from larger audience of users may bring a positive feedback on expected features which when added could lead to a robust and improved performance.

5.9 Limitations to this Study

i. The author visited many precarious and pornographic websites exposing her research instruments and tools to phishing and drive-by download attacks. Notwithstanding, computer systems and other research tools were immune with Sophos Antivirus and other updated antimalware to effectively mitigate the security threats and potential vulnerabilities.

ii. The author was conscious of the possible difficulty in accessing websites that were anonymous to VPN but this was addressed when the author's VPN was connected via an obfuscated server to disguise as if the author was using a regular internet connection.

iii. Gaining users' confidence to install and use Egenti-Filter was a difficult task for lack of trust in the application and fear of the unknown. However, an open and transparent communication was established regarding the features, advantages, and explanations of Egenti-Filter's operation; the data it gathers, and how it protects user security and privacy. This openness allayed participants' anxieties about the unknown and give them greater confidence in their choice to install Egenti-Filter. Furthermore, positive testimonials and case studies from current users were able to reinforce trust by showing real-world benefits and satisfaction of Egenti-Filter. Therefore, these strategies were effectively able to address the limitations experienced by users' lack of trust and fear of the unknown, eventually leading to higher installation rates and user satisfaction.

Nevertheless, the final output and the general integrity of the research study were unaffected by these and other limitations.

CHAPTER SIX
SUMMARY, CONCLUSION AND RECOMMENDATIONS

This final chapter is aimed at summarizing and concluding the entire work, then proffering recommendations.

6.1 SUMMARY

We have explored the myriad online security threats that users are exposed to, we have also examined a number of various security control measures and approaches. While these measures have proven effective in improving online security, they are not without some limitations. To address some of these challenges, we developed Egenti-Filter, an innovative cutting-edge model of intuitive interface that enables basic database functions.

The Egenti-Filter's performance evaluation showed an amazing accuracy rate of about 83.83%, successfully blocking access to 20 identified malicious websites while allowing legitimate content on browsers installed with the extension. Overall, the Egenti-Filter represents a significant step forward in the quest for enhanced online security, providing users with a robust tool to mitigate risks associated with malicious web content. As the digital landscape continues to evolve, ongoing development and refinement of such security measures will be crucial in safeguarding users and their online activities. The model's success in achieving the goal is evident from the results of the performance evaluation. Its adaptive nature enables it to adapt to the changing landscape of online threats, thereby blocking malicious website even before it is being accessed by the user. In terms of user feedback, the users have expressed a greater sense of security and confidence in their online activities, which has been overwhelmingly positive, confirming Egenti-Filter’s practical utility and effectiveness.

The performance of Egenti-Filter was benchmarked with the results of four studies; while the approaches differ slightly, the overarching objective of enhancing online safety is covered by all the four studies. Egenti-Filter is a lightweight Javascript, less resource intensive. It is not language specific, as it can filter non-English websites, and can be deployed everywhere irrespective of location or region.

This study has laid the groundwork for a robust web filtering model, significantly contributing to the field of online security. The positive results and user feedback confirm the model's potential impact in improving users' online activity. The findings from this study therefore, could inspire the design and development of instructional materials meant to increase public knowledge of Internet safety and the value of content filtering that improve users' rights in digital environments, and create security awareness for online users.

6.2 CONCLUSION

This development of a real-time filtering model for web browser extension has significant implications for cybersecurity resilience, user safety, and online security. Egenti-Filter is a reliable preventative security solution that proactively scrutinizes URLs before users' access potentially malicious URLs because of its strong accuracy (83.83%), precision (80.32%), and recall (96.40%) rates. The approach considerably lowers users' exposure to changing cyber-threats by utilising AI-driven threat identification to solve crucial security threats like phishing schemes, malicious infections, zero-day threats, and obfuscated URLs. Furthermore, because of its high recall, safe websites continue to be accessible, striking a balance between security and usability, which is essential for broad adoption. This study's wider implications go beyond safeguarding individual users. It has educational usefulness in increasing knowledge of online risks, enterprise-level applications in protecting corporate networks, and policy-making implications in bolstering cybersecurity frameworks. This study advances intelligent, real-time security systems that proactively protect against sophisticated cyber threats as fraudsters continue to hone their strategies. In the end, this study lays the groundwork for further developments in AI-powered cybersecurity, enhancing threat intelligence, real-time detection, and adaptive security models to make the internet a safer place for people, companies, and organisations everywhere.

6.3 RECOMMENDATIONS

The goal of research is to employ scientific methods to find answers to questions. A study process is therefore aimed at providing the researchers with the information, perspective, mindset, and expertise needed to approach issue solving from a scientific standpoint. Hence, this section enumerates the potential areas for future research as postulated by this study.

1. The use of mobile browsers to expand the Model: it is recommended that a mobile-friendly version of the browser extension that works with Chrome, Firefox, and Edge for Android and iOS without consuming too much battery life.
2. The use on-device processing for real-time URL screening that protects privacy, thereby increases protection for mobile users, who are more frequently the victim of malware and phishing attempts, and enhances security for business mobile workers using smartphones to access private information.
3. To identify questionable browsing behaviours, it is recommended to implement behavioural-based threat detection (lot of redirection, hidden pop-ups, or forced downloads), and the use heuristic analysis, which increases the ability to improve to identify social engineering scams that might not have the typical malware signatures, which will also minimise false negatives by identifying threats based on current activity rather than historical data.
4. Web filters can be more effective if cyber security professionals create a database of past and present reports and emphasize the importance of data collection during the process of web filtering. However, there is low attention to activity reporting in instances of false positives and false negatives. Unfortunately, 82% of the population claimed not to be aware of any reporting mechanisms in place. Hence, more studies are needed in this direction.

REFERENCES

Abba, A., & Hassan, M. A. (2022). Online Social Networks: Security and Privacy Issues Abubakar Abba; & Muhammed Abdulazeez Hassan Department of Computer Science, Federal College of Education, Zaria. 26(9), 87–94.

Abdallah, N., Abdalla, O., Alkhazaleh, H., & Ibrahim, A. (2020). Information security awareness behavior among higher education students: Case study. Journal of Theoretical and Applied Information Technology, 8(10), 3825–3836.

Abiade, O. (2024). TOPIC : AI-Driven Security Measures for Proactive Threat Detection Author : Oluwaseun Abiade Date : 9 th August , 2024.

Acharya, H. B., Ramesh, A., & Jalaly, A. (2019). The World from Kabul : Internet Censorship in Afghanistan. 2019–2020.

Adam, A. S. E. and A. M. E. (2019). Designing of Web Filtering Policies Ahmed Salah Eldeen and Akram Mustafa Elhaj Adam. 3(1), 11–20.

Adamides, E. D. (2023). Activity theory for understanding and managing system innovations. International Journal of Innovation Studies, 7(2), 127–141. https://doi.org/10.1016/j.ijis.2022.12.001

Adom, D., & Hussein, E. K. (2018). Theoretical and Conceptual Framework : Mandatory Ingredients International Journal of Scientific Research Theoretical and Conceptual Framework : Mandatory Ingredients Engineering Dickson Adom * Emad Kamil Hussein. January.

Ahmed, M., Khan, A., Saleem, O., & Haris, M. (2018a). A Fault Tolerant Approach for Malicious URL Filtering. 2018 International Symposium on Networks, Computers and Communications, ISNCC 2018, 1–6. https://doi.org/10.1109/ISNCC.2018.8530984

Ahmed, M., Khan, A., Saleem, O., & Haris, M. (2018b). A Fault Tolerant Approach for Malicious URL Filtering. April 2021. https://doi.org/10.1109/ISNCC.2018.8530984

Ahsan, M. M., Mahmud, M. A. P., Saha, P. K., Gupta, K. D., & Siddique, Z. (2021). Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance. Technologies, 9(3), 5–9. https://doi.org/10.3390/technologies9030052

Aktay, S. (2018). Teacher perspective on internet censorship in Turkey. Universal Journal of Educational Research, 6(2), 296–306. https://doi.org/10.13189/ujer.2018.060212

Al-Adwan, A. S., Li, N., Al-Adwan, A., Abbasi, G. A., Albelbisi, N. A., & Habibi, A. (2023). "Extending the Technology Acceptance Model (TAM) to Predict University Students' Intentions to Use Metaverse-Based Learning Platforms". Education and Information Technologies, 28(11), 15381–15413. https://doi.org/10.1007/s10639-023-11816-3

Al Shamsi, A. A. (2019). Effectiveness of Cyber Security Awareness Program for young children: A Case Study in UAE. In International Journal of Information Technology and Language Studies (IJITLS) (Vol. 3, Issue 2). http://journals.sfu.ca/ijitls

Alanazi, M., Mahmood, A., & Chowdhury, M. J. M. (2023). SCADA vulnerabilities and attacks: A review of the state-of-the-art and open issues. Computers and Security, 125. https://doi.org/10.1016/j.cose.2022.103028

Alanezi, M. (2021). Phishing Detection Methods: A Review. Technium: Romanian Journal of Applied Sciences and Technology, 3(9), 19–35. https://doi.org/10.47577/technium.v3i9.4973

Alexei, A., & Alexei, A. (2021). Cyber security threat analysis in higher education institutions as a result of distance learning. International Journal of Scientific & Technology Research, 10(3), 128–133.

Ali, F., Khan, P., Riaz, K., Kwak, D., Abuhmed, T., Park, D., & Kwak, K. S. (2019). A fuzzy ontology and SVM–based web content classification system. IEEE Access, 5, 25781–25797. https://doi.org/10.1109/ACCESS.2017.2768564

Ali, N., Khan, A., Ahmad, M., Ali, M., & Jeon, G. (2021). URL filtering using big data analytics in 5G networks. Computers and Electrical Engineering, 95(July), 107379. https://doi.org/10.1016/j.compeleceng.2021.107379

Alshaikh, M., Maynard, S. B., Ahmad, A., & Chang, S. (2018). An exploratory study of current information security training and awareness practices in organizations. Proceedings of the Annual Hawaii International Conference on System Sciences, 2018-Janua, 5085–5094. https://doi.org/10.24251/hicss.2018.635

Altarturi, H. H. M., & Anuar, N. B. (2020). A preliminary study of cyber parental control and its methods. 2020 IEEE Conference on Application, Information and Network Security, AINS 2020, 53–57. https://doi.org/10.1109/AINS50155.2020.9315134

Alzubaidi, A. (2021). Measuring the level of cyber-security awareness for cybercrime in Saudi Arabia. Heliyon, 7(1), e06016. https://doi.org/10.1016/j.heliyon.2021.e06016

Anderson, T. (2013). Challenges and Opportunities for use of Social Media in Higher Education. 6(1), 6–19.

Andriani, E., Priskananda, A. A., & Budiraharjo, M. (2022). A Cultural-Historical Activity Theory (CHAT) Analysis on Educational Psychology Class: The Challenges in Delivering a Fully Online Classroom Environment. Journal of Foreign Language Teaching and Learning, 7(1), PRESS. https://doi.org/10.18196/ftl.v7i1.13196

Angelopoulos, C. (2016). Filtering the Internet for Copyrighted Content. January 2012.

Appropriate Filtering for Education settings. (2021). https://www.saferinternet.org.uk/helpline

Athirah, N., & Suliman, B. (2017). Explicit Words Filtering Mechanism on Web Browser for Kids.

Ayubi, S., Sadat, S. A., & Jabarkhel, K. (2020). Using the Framework of Uses and Gratification Theory , an investigation of interpersonal communication on social networking sites. 07(17), 3654–3663.

Baishya, A., & Kakoty, S. (2019a). A Review on Web Content Filtering, Its Technique and Prospects. International Journal of Computer Science Trends and Technology, 7(3), 37–40. www.ijcstjournal.org

Baishya, A., & Kakoty, S. (2019b). A Review on Web Content Filtering , Its Technique and Prospects. 7(3), 37–40.

Baldry, A. C., Sorrentino, A., & Farrington, D. P. (2019). Cyberbullying and cybervictimization

versus parental supervision, monitoring and control of adolescents' online activities. Children and Youth Services Review, 96(December 2018), 302–307. https://doi.org/10.1016/j.childyouth.2018.11.058

Bandari, V. (2023). Enterprise Data Security Measures: A Comparative Review of Effectiveness and Risks Across Different Industries and Organization Types. Journal of Business Intelligence and Big Data, 6(1), 1–11. https://research.tensorgate.org/index.php/IJBIBDA/article/view/3

Banday, M. T. (2014). A Concise Study of Web Filtering. January 2008.

Banday, M. T., & Shah, N. A. (2010). A Concise Study of Web Filtering. May.

Beznazwy, J. (2019). How China Detects and Blocks Shadowsocks. 111–124.

Bhatnagar, N., & Pry, M. (2020). Student attitudes, awareness, and perceptions of personal privacy and cybersecurity in the use of social media: An initial study. In Information Systems Education Journal (ISEDJ) (Vol. 18, Issue 1). https://isedj.org/;http://iscap.info

Bhavaraju, S., Smith, T., & Zhang, B. (2018). Security Analysis of Firefox WebExtensions.

Birba, D. E. (2020). A Comparative study of data splitting algorithms for machine learning model selection. Degree Project in Computer Science and Engineering, 2020(1), 1–23. https://www.diva-portal.org/smash/get/diva2:1506870/FULLTEXT01.pdf

Bocar, A. C., & Jocson, G. G. (2022). Understanding the Challenges of Social Media Users: Management Students' Perspectives in Two Asian Countries.

Bottyán, L. (2023). Cybersecurity Awareness Among University Students. Journal of Applied Technical and Educational Sciences., 13(3), 1–11. https://doi.org/10.24368/jates363

Budiningsih, I., Soehari, T. D., & Irwansyah. (2019). Dominant factor for improving information security awareness. Cakrawala Pendidikan, 38(3), 490–498. https://doi.org/10.21831/cp.v38i3.25626

Bui, S. (n.d.). Browser Architecture.

Butun, I., Osterberg, P., & Song, H. (2020). Security of the Internet of Things: Vulnerabilities, Attacks, and Countermeasures. IEEE Communications Surveys and Tutorials, 22(1), 616–644. https://doi.org/10.1109/COMST.2019.2953364

Cai, J., Middleton, J. A., Hiebert, J., Cai, J., Hwang, S., & Morris, A. K. (2023). A New Researcher ' s Guide.

Cardoso, J., Lopes, R., & Poels, G. (2014). Conceptual frameworks. SpringerBriefs in Computer Science, 0(9783319108124), 15–33. https://doi.org/10.1007/978-3-319-10813-1_2

Carlini, N., Felt, A. P., & Wagner, D. (2012). An evaluation of the google chrome extension security architecture. Proceedings of the 21st USENIX Security Symposium, 97–111.

Chhibbar, A. (2022). Navigating the Indian Cyberspace Maze. Ashish Chhibbar.

Chrysomalidis, G. S., Teacher, S. E., Chrysomalidou, A. S., Teacher, S. E., Engineer, C., Spiliotis, I. A., Teacher, S. E., & Engineer, E. (2021). Content Filtering Technologies and the Case of Greek School Network. 9(1), 89–101.

Chyrun, L., Gozhyj, A., Yevseyeva, I., Dosyn, D., Tyhonov, V., & Zakharchuk, M. (2019). Web

content monitoring system development. CEUR Workshop Proceedings, 2362.

Clarke, B. (2019). Children And The Internet. International Journal of Advertising and Marketing to Children, 2(1), 71–75. https://doi.org/10.1108/eb027638

Cobbe, J. (2021). Algorithmic Censorship by Social Platforms: Power and Resistance. Philosophy and Technology, 34(4), 739–766. https://doi.org/10.1007/s13347-020-00429-0

Cohen Zilka, G. (2019). Awareness of eSafety and Potential Online Dangers among Children and Teenagers. Proceedings of the 2017 InSITE Conference, 16, 901–902. https://doi.org/10.28945/3683

Darer, A. P. (2020). Monitoring Internet Censorship ; Linguistic Connectivity within the Webgraph. December.

De, R., Pandey, N., & Pal, A. (2020). Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID- 19 . The COVID-19 resource centre is hosted on Elsevier Connect , the company ' s public news and information . January.

Deotte, C., Liu, B., Schifferer, B., & Titericz, G. (2021). GPU Accelerated Boosted Trees and Deep Neural Networks for Better Recommender Systems. ACM International Conference Proceeding Series, 1(1), 7–14. https://doi.org/10.1145/3487572.3487605

Desai, B., Kushwaha, U., & Jha, S. (2020). Image Filtering -Techniques , Algorithm and Applications ISSN NO : 1869-9391 Image Filtering - Techniques , Algorithm and Applications. December.

Dhillon, G., & Smith, K. J. (2019). Defining Objectives for Preventing Cyberstalking Defining Objectives for the Prevention of Cyberstalking. March. https://doi.org/10.1007/s10551-017-3697-x

Diaz-Garcia, J. A., Ruiz, M. D., & Martin-Bautista, M. J. (2022). NOFACE: A new framework for irrelevant content filtering in social media according to credibility and expertise. Expert Systems with Applications, 208(October 2021), 118063. https://doi.org/10.1016/j.eswa.2022.118063

Drăghici, G. (2022). The Internet, Lifestyle and Society. https://doi.org/10.15405/epes.23045.16

Duncan, S. P., & Chen, H. (2023). Detecting Network-based Internet Censorship via Latent Feature Representation Learning.

Egenti, Grace; Nwaocha, Vivian; Olalere, O. J. (2023). Investigating the level of information security awareness amongst Nigerian tertiary institutions □. 060025.

Etuh, E., S. Bakpo, F., & A.H, E. (2021). Social Media Network Attacks and their Preventive Mechanisms: A Review. 59–74. https://doi.org/10.5121/csit.2021.112405

Faruk, J. H., Shahriar, H., Valero, M., Barsha, F. L., & Sobhan, S. (2021). Malware Detection and Prevention using Artificial Intelligence Techniques. February 2022. https://doi.org/10.1109/BigData52589.2021.9671434

Fatichah, C., Lazuardi, W. F., Navastara, D. A., Suciati, N., & Munif, A. (2021). A Content Filtering from Spam Posts on Social Media using Weighted Multimodal Approach. 2012. https://doi.org/10.3844/jcssp.2021.55.66

Federal Communicatiions Commission. (2020). Cyber Security Planning Guide.

Figueira, G., Barradas, D., & Santos, N. (2022). Stegozoa: Enhancing Web RTC Covert Channels with Video Steganography for Internet Censorship Circumvention. ASIA CCS 2022 - Proceedings of the 2022 ACM Asia Conference on Computer and Communications Security, 1154–1167. https://doi.org/10.1145/3488932.3517419

Furnell, S., & Vasileiou, I. (2017). Security education and awareness: just let them burn? Network Security, 2017(12), 5–9. https://doi.org/10.1016/S1353-4858(17)30122-8

Garba, A. A., Siraj, M. M., & Othman, S. H. (2022). An assessment of cybersecurity awareness level among Northeastern University students in Nigeria. International Journal of Electrical and Computer Engineering, 12(1), 572–584. https://doi.org/10.11591/ijece.v12i1.pp572-584

Giovanni Sartor and Andrea Loreggia. (2020). The impact of algorithms for online content filtering or moderation. September, 72.

Gogus, A. (2023). Adaptation of an Activity Theory Framework for Effective Online Learning Experiences: Bringing Cognitive Presence, Teaching Presence, and Social Presence to Online Courses. Online Learning Journal, 27(2), 265–287. https://doi.org/10.24059/olj.v27i2.3073

Gómez Hidalgo, J. M., Sanz, E. P., García, F. C., & Rodríguez, M. D. B. (2019). Chapter 7 Web Content Filtering. Advances in Computers, 76(09), 257–306. https://doi.org/10.1016/S0065-2458(09)01007-9

Gupta, N., & Hilal, S. (2019). Algorithm to filter & redirect the web content for kids'. International Journal of Engineering and Technology, 5(1), 88–94.

Hagger, M. S. (2019). The Reasoned Action Approach and the Theories of Reasoned Action and Planned Behavior. Psychology, May. https://doi.org/10.1093/obo/9780199828340-0240

Hao, J., & Ho, T. K. (2019). Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language. Journal of Educational and Behavioral Statistics, 44(3), 348–361. https://doi.org/10.3102/1076998619832248

Hashim, N., & Jones, M. L. (2007). Activity Theory: A framework for qualitative analysis. 4th International Qualitative Research Convention (QRC), September.

Hausknecht, D., Magazinius, J., & Sabelfeld, A. (2015). May I?-content security policy endorsement for browser extensions. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9148(March), 261–281. https://doi.org/10.1007/978-3-319-20550-2_14

Herath, T. B. G., Khanna, P., & Ahmed, M. (2022). Cybersecurity Practices for Social Media Users: A Systematic Literature Review. Journal of Cybersecurity and Privacy, 2(1), 1–18. https://doi.org/10.3390/jcp2010001

Hoang, N. P., Niaki, A. A., Dalek, J., Knockel, J., Lin, P., Marczak, B., Polychronakis, M., & Symposium, U. S. (2021). How Great is the Great Firewall ? Measuring China ' s DNS Censorship.

Hohensee, J. H. · J. C. · S. H. · A. K. M. · C. (2023). A New Researcher ' s Guide.

Hounsel, A., Mittal, P., & Feamster, N. (2019). Automatically generating a large, culture-specific

blocklist for China. 8th USENIX Workshop on Free and Open Communications on the Internet, FOCI 2018, Co-Located with USENIX Security 2018.

Hyeon, J. (2023). Predictors of social networking service addiction. Scientific Reports, 1–15. https://doi.org/10.1038/s41598-023-43796-2

Ibrahim, A. A. (2019). Definition Purpose and Procedure of Developmental Research : An Analytical Definition Purpose and Procedure of Developmental Research : An Analytical Review. August. https://doi.org/10.9734/ARJASS/2016/30478

Internetsociety.org. (2017). Internet Society Perspectives on Internet Content Blocking: An Overview. March. https://www.internetsociety.org/resources/doc/2017/internet-content-blocking/

ITU. (2009). Guidelines for Parents , Guardians and Educators on Child Online Protection.

ITU. (2020). Guidelines for Parents , Guardians and Educators on Child Online Protection.

Jain, A. K., Sahoo, S. R., & Kaubiyal, J. (2021). Online social networks security and privacy: comprehensive review and analysis. Complex and Intelligent Systems, 7(5), 2157–2177. https://doi.org/10.1007/s40747-021-00409-7

Jain, P. (2018). Survey on Web Browser and Their Extensions 1. 2(2), 41–55.

Jamali, H. R. (2018). The effects of internet filtering on users ' information-seeking behaviour and emotions. February. https://doi.org/10.1108/AJIM-12-2016-0218

Jijo, B. T., & Abdulazeez, A. M. (2021). Classification Based on Decision Tree Algorithm for Machine Learning. 02(01), 20–28. https://doi.org/10.38094/jastt20165

Jillepalli, A. A., De Leon, D. C., Steiner, S., Sheldon, F. T., & Haney, M. A. (2017). Hardening the client-side: A guide to enterprise-level hardening of web browsers. Proceedings - 2017 IEEE 15th International Conference on Dependable, Autonomic and Secure Computing, 2017 IEEE 15th International Conference on Pervasive Intelligence and Computing, 2017 IEEE 3rd International Conference on Big Data Intelligence and Compu, 2018-Janua(April 2021), 687–692. https://doi.org/10.1109/DASC-PICom-DataCom-CyberSciTec.2017.120

Johanna, H. (2019). Information security awareness and behavior. Management Research Review, 37(12), 1049–1092. https://doi.org/10.1108/MRR-04-2013-0085

Johnson, C., Khadka, B., Basnet, R. B., & Doleck, T. (2020). Towards Detecting and Classifying Malicious URLs Using Deep Learning. 11(4), 31–48. https://doi.org/10.22667/JOWUA.2020.12.31.031

Joseph, V. R., & Vakayil, A. (2022). SPlit: An Optimal Method for Data Splitting. Technometrics, 64(2), 166–176. https://doi.org/10.1080/00401706.2021.1921037

Junaid-ur-Rehman, S. (2022). Evaluating the Effectiveness of Reasoned-action Theories (TRA, TPB, IBM) for Explaining Low E-commerce Adoption in a Developing Country: A Structural Equation Modelling (SEM) approach AI trends in digital humanities research. Trends in Computer Science and Information Technology, 7(2), 035–046. https://doi.org/10.17352/tcsit.000049

Kaddoura, S., Chandrasekaran, G., Popescu, D. E., & Duraisamy, J. H. (2022). A systematic literature review on spam content detection and classi fi cation.

https://doi.org/10.7717/peerj-cs.830

Kannelønning, K., & Katsikas, S. K. (2023). A systematic literature review of behavior has been assessed. 310105. https://doi.org/10.1108/ICS-08-2022-0139

Kareem Thajeel Thajeel, I., Samsudin, K., Jahari Hashim, S., & Hashim, F. (2023). Machine and Deep Learning-based XSS Detection Approaches: A Systematic Literature Review. Journal of King Saud University - Computer and Information Sciences, 35, 101628. https://doi.org/10.1016/j.jksuci.2023.101628

Karim, R. (2015). Techniques and Tools for Secure Web Browser Extension Development.

Kashmar, N., Adda, M., Atieh, M., & Ibrahim, H. (2020). Deriving access control models based on generic and dynamic metamodel architecture: Industrial use case. Procedia Computer Science, 177, 162–169. https://doi.org/10.1016/j.procs.2020.10.024

Kashmar, N., Adda, M., & Ibrahim, H. (2022). Access Control Metamodels: Review, Critical Analysis, and Research Issues. Journal of Ubiquitous Systems and Pervasive Networks, 16(2), 93–102. https://doi.org/10.5383/juspn.16.02.006

Kashmar, N., Adda, M., Ibrahim, H., & Atieh, M. (2021). Access Control in Cybersecurity and Social Access Control in Cybersecurity. February.

Kaur, R., & Singh, E. R. (2017). Image Filtering Techniques-A Review.

Khader, M., Karam, M., & Fares, H. (2021). Cybersecurity awareness framework for academia. Information (Switzerland), 12(10). https://doi.org/10.3390/info12100417

Khan, U., Sarim, M., Ahmad, M. Bin, & Shafiq, F. (2019). Feature extraction and modeling techniques in speech recognition: A review. Proceedings - 2019 4th International Conference on Information Systems Engineering, ICISE 2019, IX(Ii), 63–67. https://doi.org/10.1109/ICISE.2019.00020

Khandkar, V. S., & Hanawal, M. K. (2021). Masking Host Identity on Internet : Encrypted TLS / SSL Handshake.

Khando, K., Gao, S., Islam, S. M., & Salman, A. (2021a). Enhancing employees information security awareness in private and public organisations : A systematic literature review. Computers & Security, 106, 102267. https://doi.org/10.1016/j.cose.2021.102267

Khando, K., Gao, S., Islam, S. M., & Salman, A. (2021b). Enhancing employees information security awareness in private and public organisations: A systematic literature review. Computers and Security, 106, 102267. https://doi.org/10.1016/j.cose.2021.102267

Khder, M. A. (2021). Web scraping or web crawling: State of art, techniques, approaches and application. International Journal of Advances in Soft Computing and Its Applications, 13(3), 144–168. https://doi.org/10.15849/ijasca.211128.11

Khoo, L. M. S., Chieu, H. L., Qian, Z., & Jiang, J. (2020). Interpretable rumor detection in microblogs by attending to user interactions. AAAI 2020 - 34th AAAI Conference on Artificial Intelligence, 8783–8790. https://doi.org/10.1609/aaai.v34i05.6405

Koohang, A. (2021). Social media sites privacy concerns: Empirical validation of an instrument. Online Journal of Applied Knowledge Management, 5(1), 14–26. https://doi.org/10.36965/ojakm.2017.5(1)14-26

Kovaˇ, A. (2020). applied sciences SAWIT — Security Awareness Improvement Tool in the Workplace.

Kozyreva, A., Lewandowsky, S., & Hertwig, R. (2020). Citizens Versus the Internet : Confronting Digital Challenges With Cognitive Tools. https://doi.org/10.1177/1529100620946707

Krasnova, K. (2022). Społeczeństwo , polityka ,. September. https://doi.org/10.31648/pw.3001

Kumar, R., Kumar, D. P., & Kumar, D. V. (2022). Design And Implementation Of Privacy And Security System In Social Media. International Journal of Advanced Networking and Applications, 13(04), 5081–5088. https://doi.org/10.35444/ijana.2022.13411

Kumari, R., Ashok, N., Ghosal, T., & Ekbal, A. (2021). Misinformation detection using multitask learning with mutual learning for novelty detection and emotion recognition. Information Processing and Management, 58(5), 102631. https://doi.org/10.1016/j.ipm.2021.102631

Lausanne. (2017). On Blocking , Filtering and Take-Down of Comparative Study on Blocking , Filtering and Take-Down Of Illegal.

Lee, G. W., & Kim, H. K. (2020). applied sciences Multi-Task Learning U-Net for Single-Channel Speech Enhancement and Mask-Based Voice Activity Detection.

Maalem Lahcen, R. A., Caulkins, B., Mohapatra, R., & Kumar, M. (2020). Review and insight on the behavioral aspects of cybersecurity. Cybersecurity, 3(1). https://doi.org/10.1186/s42400-020-00050-w

Malami, A. (2020). Federal Ministry of Justice.

Mamun, M., Canada, C., Rathore, M. A., Lashkari, A. H., & Stakhanova, N. (2016). Detecting Malicious URLs Using Lexical. September. https://doi.org/10.1007/978-3-319-46298-1

Maple, C. (n.d.). The Impact of Cyberstalking : review and analysis of the ECHO Pilot Project.

Maserumule, N. T. (2020). Parent's Use of Strategies to Monitor Children's Activities Online. Open Journal of Social Sciences, 08(05), 506–536. https://doi.org/10.4236/jss.2020.85034

Mathews, N. S., & Chimalakonda, S. (2021). Detox Browser - Towards Filtering Sensitive Content On the Web. 1(1), 1–6.

Michalopoulos, D., Mavridis, I., & Jankovic, M. (2014). GARS: Real-time system for identification, assessment and control of cyber grooming attacks. Computers and Security, 42, 177–190. https://doi.org/10.1016/j.cose.2013.12.004

Mienye, I. D., & Sun, Y. (2022). A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects. IEEE Access, 10, 99129–99149. https://doi.org/10.1109/ACCESS.2022.3207287

Morrow, D., & Morrow, D. D. (2022). Chapman University Digital Commons Modern American Propaganda : An Institutional History Modern American Propaganda : An Institutional History A Thesis by.

Mugisha, D. (2019). Web Browser Forensics : Evidence collection And Analysis for Most Popular Web Browsers usage in Windows 10 Gujarat Forensic Sciences University Institute of Forensic Science M . Sc . Digital Forensics and Information Security Web Browser

Forensics : Evide. September 2018. https://doi.org/10.13140/RG.2.2.25857.51049

Muraina, I. O. (2022). Ideal Dataset Splitting Ratios in Machine Learning Algorithms: General Concerns for Data Scientists and Data Analysts. 7th International Mardin Artuklu Scientific Researches Conference, February, 496–504. https://www.researchgate.net/publication/358284895_Ideal_Dataset_Splitting_Ratios_In_Machine_Learning_Algorithms_General_Concerns_For_Data_Scientists_And_Data_Analysts

Nagar, G., & Manoharan, A. (2024). Understanding the Threat Landscape: a Comprehensive Analysis of Cyber-Security Risks in 2024. March. www.irjmets.com

Naqvi, B., Perova, K., Farooq, A., Makhdoom, I., Oyedeji, S., & Porras, J. (2023). Computers & Security Mitigation strategies against the phishing attacks : A systematic literature review. Computers & Security, 132, 103387. https://doi.org/10.1016/j.cose.2023.103387

Narwal, N. (2018). Web Content Filtering using Machine Learning Approach. 2.

Narwal, N. (2021). Web page filtering for kids. International Journal of Information Technology (Singapore), 13(1), 19–25. https://doi.org/10.1007/s41870-020-00474-0

Nash, A. K. P. & V. (2021). Internet Filtering Technology and Aversive Online Experiences in Adolescents.

Nawalagatti, A., State, K., Development, R., & Raj, P. (2022). Analysis of Security and Privacy. International Journal of Creative Research Thoughts (IJCRT), 10(3), 419–423.

Nawaz, N. A., Ishaq, K., Farooq, U., Khalil, A., Rasheed, S., Abid, A., & Rosdi, F. (2022). Computer Science Manuscript to be reviewed A Comprehensive Review of Security Threats and.

Nawaz, N. A., Ishaq, K., Farooq, U., Khalil, A., Rasheed, S., Abid, A., & Rosdi, F. (2023). A comprehensive review of security threats and solutions for the online social networks industry. PeerJ Computer Science, 9, e1143. https://doi.org/10.7717/peerj-cs.1143

NCC. (2018). Final report on effects of Cyber Crime on Foreign Direct Investment and National Development. Newark Security System.

Nguyen, Q. H., Ly, H. B., Ho, L. S., Al-Ansari, N., Van Le, H., Tran, V. Q., Prakash, I., & Pham, B. T. (2021). Influence of data splitting on performance of machine learning models in prediction of shear strength of soil. Mathematical Problems in Engineering, 2021. https://doi.org/10.1155/2021/4832864

Nicolaidou, I., & Venizelou, A. (2020). Improving children's E-safety skills through an interactive learning environment: A quasi-experimental study. Multimodal Technologies and Interaction, 4(2). https://doi.org/10.3390/mti4020010

Nicoletti, P. (2013a). Content Filtering. In Computer and Information Security Handbook (Vol. 9780123943). Elsevier Inc. https://doi.org/10.1016/B978-0-12-394397-2.00066-0

Nicoletti, P. (2013b). Content Filtering. In Computer and Information Security Handbook 2e. Elsevier Inc. https://doi.org/10.1016/B978-0-12-394397-2.00093-3

Nikolaev, I., Grill, M., & Valeros, V. (2015). Exploit Kit Website Detection Using HTTP Proxy Logs. 2–7.

Norma Guti´errez. (2021). Malicious websites blocking system using Deep Learning algorithms

Project Thesis.

Odiaga, G. A., Abeka, S., & Liyala, S. (2020). An Information Security Awareness Framework For Secondary School Teachers In Kenya. May. http://62.24.102.115:8080/handle/123456789/8944%0Ahttp://62.24.102.115:8080/bitstream/handle/123456789/8944/Odiaga_ An Information Security Awareness Framework For Secondary School Teachers In Kenya.pdf?sequence=1&isAllowed=y

Ogundele, I. O., Akinade, A. O., & Alakiri, H. O. (2020). Detection and Prevention of Session Hijacking in Web Application Management. Ijarcce, 9(7), 1–10. https://doi.org/10.17148/ijarcce.2020.9601

Ojo, F. (2024). An Overview of Web Content Filtering Techniques BY OJO , FOLORUNSO FIDELIS Department of Computer Science Federal College of Education , Abeokuta Tel No : 08065797801. June.

Okesola, J. O. (2014). Measuring information security awareness effectiveness in social networking sites – a non-incident statistics approach by Julius Olatunji OKESOLA Submitted in accordance with the requirements for the degree of Doctor of Philosophy in University of South A. December.

Omeluzor, S. U., Okonoko, V. N., & Anene, O. E. (2023). Technologies for recovery and growth in post COVID-19 era in tertiary institutions in Nigeria. 20, 1–12. https://doi.org/10.1016/j.sciaf.2023.e01602

On, T. O., Filtering, I. S. P. B., & Internet, O. F. T. H. E. (2019). ISP BASED FILTERING COMPUTER SOCIETY ICT Professionals Shaping Our Future About the Australian Computer Society COMPUTER.

OSUJI, U. S. A. (PhD, & ISHOLA, T. O. (PhD). (2012). Course code: fms 304 course title: research methodology 1. 1–188. https://nou.edu.ng/coursewarecontent/FMS 304 RESEARCH METHODOLOGY_0.pdf

Overhaul, A. (2022). FREEDOM ON THE NET 2022 Countering an Authoritarian Overhaul of the Internet FREEDOM ON THE NET 2022.

Ozbay, F. A., & Alatas, B. (2020). Fake news detection within online social media using supervised artificial intelligence algorithms. Physica A: Statistical Mechanics and Its Applications, 540, 123174. https://doi.org/10.1016/j.physa.2019.123174

Ozimek, P., & Förster, J. (2021). The Social Online-Self-Regulation-Theory: A Review of Self-Regulation in Social Media. Journal of Media Psychology, 33(4), 181–190. https://doi.org/10.1027/1864-1105/a000304

Perera, S. (2021). Investigation of social media security : A Critical Review. Department of Information Technology Sri Lanka Institute of Information Technology, 4(March), 1–5. https://www.researchgate.net/publication/349944503

Persia, F., & Auria, D. D. (2017). A Survey of Online Social Networks : Challenges and Opportunities University of Naples. Figure 1. https://doi.org/10.1109/IRI.2017.74

Peter J. R. Macaulay, Michael J. Boulton, Lucy R. Betts, Louise Boulton, Eleonora Camerone, James Down, Joanna Hughes, C. K. & R. K. (2019). Children’s Knowledge Of Online Safety/Danger. 1–43.

Petru-cristian, N. (2023). A Comprehensive Analysis of High-Impact Cybersecurity Incidents: Case Studies and Implications. October. https://doi.org/10.13140/RG.2.2.17461.65763

Postholm, M. B. (2015). Methodologies in Cultural–Historical Activity Theory: The example of school-based development. Educational Research, 57(1), 43–58. https://doi.org/10.1080/00131881.2014.983723

Potgieter, P. (2019). The Awareness Behaviour of Students on Cyber Security Awareness by Using Social Media Platforms: A Case Study at Central University of Technology. 12, 272–262. https://doi.org/10.29007/gprf

Quayyum, F., Cruzes, D. S., & Jaccheri, L. (2021). Cybersecurity awareness for children: A systematic literature review. In International Journal of Child-Computer Interaction (Vol. 30). Elsevier B.V. https://doi.org/10.1016/j.ijcci.2021.100343

Ramalingam, R., Khan, S., & Mohammed, S. (2016). The need for effective information security awareness practices in Oman higher educational institutions. May. http://arxiv.org/abs/1602.06510

Raman, R. S., Shenoy, P., Kohls, K., & Ensafi, R. (2020). Censored Planet : An Internet-wide , Longitudinal Censorship Observatory. 49–66.

Raman, R. S., Stoll, A., Dalek, J., Ramesh, R., Scott, W., & Ensafi, R. (2020). Measuring the Deployment of Network Censorship Filters at Global Scale. February.

Ramesh, R., Raman, R. S., Bernhard, M., & Ongkowijaya, V. (2020). Decentralized Control : A Case Study of Russia. February.

Rapacz, S., Chołda, P., & Natkaniec, M. (2021). A Method for Fast Selection of Machine-Learning Classifiers for.

Rask, J. K., Madsen, F. P., Battle, N., & Macedo, H. D. (2020). Visual Studio Code VDM Support. December.

Rathore, S., Kumar, P., Loia, V., Jeong, Y., & Hyuk, J. (2017). Social network security : Issues , challenges , threats , and solutions. Information Sciences, 421, 43–69. https://doi.org/10.1016/j.ins.2017.08.063

Safe, K. C. (2022). Appropriate Filtering for Education settings. June.

Salawu Rafiu Oyesola, A.-O. S. B. and S. M. (2023). Theoretical and Conceptual Frameworks In Research: Conceptual Clarification Section A-Research paper. 12(12), 2103–2117.

Saliu, H., Rexhepi, Z., Shatri, S., & Kamberi, M. (2022). Experiences With and Risks of Internet Use Among Children in Kosovo. Izobraževanje Journal of Elementary Education, 15(2), 145–164. https://doi.org/10.18690/rei.15.2.145-164.2022

Salloum, S. A., Al-Emran, M., Habes, M., Alghizzawi, M., Ghani, M. A., & Shaalan, K. (2021). What impacts the acceptance of e-learning through social media? An empirical study. In Studies in Systems, Decision and Control (Vol. 335, Issue March). Springer International Publishing. https://doi.org/10.1007/978-3-030-64987-6_24

Santiago. (2021). Digital technologies for a new future.

Sasaki, Y., & Hobbs, J. (2019). Internet safety. Encyclopedia of Cyber Behavior, 1(May), 960–974. https://doi.org/10.4018/978-1-4666-0315-8.ch079

Shen, S., Xu, K., Sotiriadis, M., & Wang, Y. (2020). Exploring the factors influencing the adoption and usage of Augmented Reality and Virtual Reality applications in tourism education within the context of COVID-19 pandemic. January.

Sheth, J. (2020). Impact of Covid-19 on consumer behavior : Will the old habits return or die ? Journal of Business Research, 117, 280–283. https://doi.org/10.1016/j.jbusres.2020.05.059

Shikalepo, E. E. (2020). Defining a conceptual framework in educational research defining. Namibia University of Science and Technology, 1(2), 1–8. https://doi.org/10.13140/RG.2.2.26293.09447

Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2020). FakeNewsNet : A Data Repository with News Content , Social Context , and Spatiotemporal Information for Studying Fake News on Social Media. 8(3). https://doi.org/10.1089/big.2020.0062

Simon Kemp. (2023). Digital 2023 Global Overview Report the Essential Guide To The World's Connected Behaviours.

SINGH, K. (2022). On the Implementation of Internet Censorship in Indiau nder the supervision of Dr . Sambuddho Chakravarty Master of Technology.

Smith, D. T. (2019). You 've Been Hacked : a Technique for Raising. Issues in Information Systems, 20(1), 186–194. https://iacis.org/iis/2019/1_iis_2019_186-194.pdf

Sonicwall. (2020). Comparing Architectures For Internet Content Filtering Solutions Content filtering : its rising importance. 1–14.

Steingr\'\imsson, S., Loftsson, H., & Way, A. (2023). Filtering Matters: Experiments in Filtering Training Sets for Machine Translation. Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa), 588–600. https://aclanthology.org/2023.nodalida-1.58

Tambe, U. S., Kakad, N. R., Suryawanshi, S. J., & Bhamre, S. S. (2021). Content Filtering of Social Media Sites Using Machine Learning Techniques. 0. https://doi.org/10.3233/APC210226

Thangaraj, M. (2019). KT - Grand : An Algorithm for Web Content Filtering International Journal of Advance Research in. 371–376.

Turner, B. R. (2021). Edith Cowan University.

Turner, B. R. (2022). An investigation into the efficacy of URL content filtering systems. The Grants Register 2023, 432–433. https://doi.org/10.1057/978-1-349-96053-8_427

Umar, I. (2020). The impact of cybercrime on the Nigerian economy and banking system. International Journal of Innovative of Scienec and Research Technology.

Van Hasselt, V. B., & Bourke, M. L. (2018). Handbook of behavioral criminology. Handbook of Behavioral Criminology, 2012, 1–762. https://doi.org/10.1007/978-3-319-61625-4

Vasavi, B. (2014). Sign-on Mechanism for Distributed Computer Networks. 2(3), 439–445.

Veli, F. A. T. (2019). Interactive Technology and Smart Education Article information : Risks Awareness and E-Safety Needs of Children.

Venugeetha, Y., Rathod, R., & Kumar, R. (2022). Social Networking Sites in Daily Life:

Benefits and Threats. Artificial Intelligence and Communication Technologies, June, 51–64. https://doi.org/10.52458/978-81-955020-5-9-5

Ververis, V., Ermakova, T., Isaakidis, M., Basso, S., Fabian, B., & Milan, S. (2021). Understanding Internet Censorship in Europe: The Case of Spain. ACM International Conference Proceeding Series, 319–328. https://doi.org/10.1145/3447535.3462638

Vivekanandam, B., & Midhunchakkaravarthy. (2022). Preventive Measures for the Impacts of Social Media Networks in Security and Privacy - A Review. Journal of ISMAC, 3(4), 291–300. https://doi.org/10.36548/jismac.2021.4.001

Vondersaar, L. (2020). Impact of Internet Content Filtering on Advanced Academics English Language Arts. July.

Voros, T., Bergeron, S. P., & Berlin, K. (2023). Web Content Filtering Through Knowledge Distillation of Large Language Models. Proceedings - 2023 22nd IEEE/WIC International Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT 2023, 357–361. https://doi.org/10.1109/WI-IAT59888.2023.00058

Wong, S. H. W., & Liang, J. (2021). Dubious until officially censored: Effects of online censorship exposure on viewers' attitudes in authoritarian regimes. Journal of Information Technology and Politics, 18(3), 310–323. https://doi.org/10.1080/19331681.2021.1879343

Xuan, C. Do, & Nguyen, H. D. (2020). Malicious URL Detection based on Machine Learning. 11(1), 148–153.

Zahidah, Z., Nurul Nuha Abdul, M., Nurul Hayani, A. R., & Shuhaili, T. (2020). Cyber Security Awareness Among Secondary School Students in Malaysia. Journal of Information Systems and Digital Technologies, 2(2), 28–41.

Zeebaree, S. R. M., Ameen, Y., & Sadeeq, M. A. M. (2020). Social Media Networks Security Threats, Risks and Recommendation: A Case Study in the Kurdistan Region. International Journal of Innovation, Creativity and Change. Www.Ijicc.Net, 13(7), 1–18. www.ijicc.net

Zhang, J., Zhang, Y., & Xu, F. (2019). Does cognitive-behavioral therapy reduce internet addiction? Protocol for a systematic review and meta-analysis. Medicine (United States), 98(38), 1–4. https://doi.org/10.1097/MD.0000000000017283

Zhang, X., Chen, Y., Hu, L., & Wang, Y. (2022). The metaverse in education: Definition, framework, features, potential applications, challenges, and future research topics. Frontiers in Psychology, 13(October), 1–18. https://doi.org/10.3389/fpsyg.2022.1016300

Zittrain, J. L., Faris, R., Noman, H., Clark, J., Tilton, C., & Morrison-Westphal, R. (2017). The Shifting Landscape of Global Internet Censorship. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.2993485

Zwilling, M., Klien, G., Lesjak, D., Wiechetek, Ł., Cetin, F., & Basim, H. N. (2022a). Cyber Security Awareness, Knowledge and Behavior: A Comparative Study. Journal of Computer Information Systems, 62(1), 82–97. https://doi.org/10.1080/08874417.2020.1712269

Zwilling, M., Klien, G., Lesjak, D., Wiechetek, Ł., Cetin, F., & Basim, H. N. (2022b). Cyber Security Awareness, Knowledge and Behavior: A Comparative Study. Journal of Computer Information Systems, 62(1), 82–97. https://doi.org/10.1080/08874417.2020.1712269

APPENDICES

APPENDIX A

NOUN RESEARCH ETHICS APPROVED CHECKLIST

The Checklist is designed to identify the nature of any ethical issues raised by the research. This checklist must be completed before potential participants are approached to take part in any research	
1. Name of Researcher: EGENTI Grace Ebere	
Please tick appropriately	Undergraduate student [ ] Masters student [ ] Ph.D. student [✔] Staff[ ]
2. Contacts	
e-mail: egentig@gmail.com	**Phone:** +234 8028411464
Department: Cybersecurity	
3. Student Details if applicable	
Degree program: PhD	
Supervisor's name: Prof. Okesola Julius	
Supervisor's e-mail: olatunjiokesola@gmail.com	
4. Title of the Thesis and brief Abstract	
Development of an Improved Web Browser Extension Filter for Online Security	
The surge in digital activities reinforces the need for the protection of online users from malicious, unsafe and unwanted content as well as data breaches. With such a need, web content filtering solutions have evolved as an efficient way of ensuring online security by regulating web access through rules and policies, and other approaches, such as blocking undesirable websites. These techniques help in protecting sensitive data, increasing employee productivity output, and improving user experience through minimization and in some cases elimination of exposure to dangerous online material. Ensuring a safe web environment is still a challenge, as new threats evolve and the amount of new content created each day is quite high. The problem lies within the structure of many existing systems and the approach to users: the focus is primarily placed on adequate protection with little relevance accorded to user experience. Often times, online security relies heavily on filtering which fails to show desired results in times when urgently needed (real-time). Even so, different approaches that include	

the use of machine learning and database systems have been explored in previous studies to improve web content filtering methods. As much as these approaches claim to be useful, such researchers still maintain that their work does not provide the level of scalability, accuracy, customisation and real-time performance that is needed in dynamic web environments. Some of the identified weaknesses in the previous solutions relate to the capability to adequately handle and process large volumes of web traffic, manually updating the database, the failure to take into account the end user's personalization aspects, as well as the challenges of content over-blocking and under-blocking. Moreover, the majority of the already developed solutions were built with a reactive stance rather than an anticipatory one, therefore, not so helpful in preventing new threats that emerges on a daily basis. This study developed a real-time web content filtering model, known as Egenti-Filter using the developmental research design methodology. It utilizes artificial intelligence to improve the model's flexibility, scalability and user-oriented approach. With Egenti-Filter some of these shortcomings were addressed, thus increasing users' online security and improving users' experiences by intercepting URLs containing malware even before the users come into contact with the URLs. After deployment, a few phishing websites were accessed, and they were blocked by the browser extension, therefore confirming the usefulness of the developed browser extension, which offers a proactive and efficient approach for safeguarding online activities in a transforming cyber-space. With an intuitive Graphical User Interface, users can easily set their filtering preferences and reconfigure controls. With this interface, users can more easily redefine filtering schemas based on their preferences or internet policies. The web content filter ensures that users could timely and effectively navigate and manage their online activities in accordance with the defined rules by providing such a degree of customisation. This versatility increases user experience and simultaneously enhances the strength of the filter in the prevention of accessing malicious URLs. An accuracy rate of 83.83% was achieved with our model.

Keywords: browser extension, digital activities, over-blocking, real-time, under-blocking, web content filtering

5.	**Funding**
Is it proposed that the research will be funded? If so by whom? Yes, by ACETEL	
6.	**Where will the research be conducted**
In what country/ies will the research take place:	Nigeria
7	Please tick in the appropriate right-hand column/box

	Yes	**NO**
Research that may need to be reviewed by an external (non-NOUN) Ethics Committee		✔
Will the study require Health Research Authority approval for use of animals genetically modified organisms (GMOs), radiation or such sensitive materials?		✔
Does the study involve participants lacking capacity to give informed consent?		✔
Is there any other reason why the study may need to be reviewed by another		
external (non-NOUN) Ethics Committee? If yes, please give details here		✔

8 Consent		
Does the study involve children or other participants who are potentially or in any way vulnerable or who may have any difficulty giving meaningful consent to their participation or the use of their information		✔
Are subjects to be involved in the study without their knowledge and consent (e.g. through internet-mediated research, or via covert observation of people in public places)?		✔
Will the study require the co-operation of a gatekeeper for initial access to the groups or individuals to be recruited? (Answer 'yes' to this question only if the involvement of a gatekeeper in your study might raise issues of whether participants' involvement is truly voluntary or of whether the gatekeeper might influence potential participants in some other way.)		✔
9. Research Design / Methodology		
Does the research methodology involve the use of deception		✔
Are there any significant concerns regarding the design of the research project? For example: • where research intrudes into the private sphere or delves into some deeply personal experience; • where the study is concerned with deviance or social control; • where the study impinges on the vested interests of powerful persons or the exercise of coercion or domination; or • where the research deals with things that are sacred to those being studied that they do not wish profaned.		 ✔ ✔ ✔ ✔
Does the proposed research relate to the provision of social or human services		✔
10. Financial Incentives		
Will financial inducements (other than reasonable expenses and compensation for time) be offered to participants that might have an impact on the objectivity of the research?		✔
11. Research Subjects		
Could the study induce unacceptable psychological stress or anxiety or cause harm or negative consequences beyond the risks encountered in normal life?		✔
Will the study involve discussion of sensitive topics? For example, (but not limited to): sexual activity, illegal behaviour, experience of violence or abuse, drug use, etc.).		✔
Are drugs, placebos or other substances to be administered to study participants or will the study involve invasive, intrusive or potentially harmful procedures of any kind?		✔
12. Confidentiality		
Will research involve the sharing of data or confidential information beyond the initial consent given?		✔
Is there ambiguity about whether the information/data you are collecting is considered to be public?		✔

iii Will the research involve administrative or secure data that requires permission from the appropriate authorities before use?		✔
Will the research involve the use of visual/vocal methods that potentially pose an issue regarding confidentiality and anonymity?		✔
13. Legal requirements		
Is there any reason why the research will NOT comply with the requirements of current data protection legislation (NITDA Data Protection act of 2019)		✔
14 Dissemination		
Are there any particular groups who are likely to be harmed by dissemination of the results of this project? Or is there any potential for misuse of the findings?		✔
15. Risk to researchers		
Does your research pose any risks to your physical or psychological wellbeing, or that of others working with you?		✔
16 Sensitive research materials		
Will the research involve accessing security-sensitive material, such as material related to terrorism or violent extremism of any kind?		✔

RESEARCHER'S SIGNATURE AND DATE:  **9th September 2023**

SIGNATURE OF ETHICS COMMITTEE CHAIRMAN AND APPROVAL/DATE: …….
NAME/SIGNATURE OF ANY OTHER APPROVING AUTHORITY: ……

APPENDIX B

QUESTIONNAIRE ON

DEVELOPMENT OF AN IMPROVED WEB BROWSER EXTENSION FILTER FOR ONLINE SECURITY

Dear Prospective Respondent,

I am a doctoral candidate on a PhD programme at the Africa Centre of Excellence on Technology Enhanced Learning (ACETEL), of the National Open University of Nigeria (NOUN). I am carrying out a survey on the title, '**Developing an Improved Web Browser Extension Filtering Model for Online Security**'. Please be assured that any information provided by you will be strictly for research purpose and will be treated with absolute confidentiality.

INFORMED CONSENT

Please be fully informed as follows:

1. **Right to Participate:** Your response is of utmost importance to this exercise, however, you have the right to and not to participate. Your participation is voluntary and you are not under any obligation to oblige**.**
2. **Right to Withdraw:** We will appreciate your response to all the questions in the Questionnaire. However, you are required to determine the number and type of questions you are comfortable with. You also have an exclusive right to discontinue at any stage of this exercise without notice.
3. **Right to Give Informed Consent**: You are not required in any way to give an informed consent to withdraw or not to participate. However, responding to the questions in the Questionnaire is the best form of informed consent we can ever have.
4. **Right to Anonymity:** You are not required to disclose your personal identity in this exercise. You may remain anonymous and still attend to the questionnaire adequately. In the event that your personal identity is disclosed, you can be rest assured that the information you provide is strictly for research purpose and will be treated with absolute confidence and anonymity.
5. **Right to Confidentiality:** the essence of this exercise is Securing Online Activities: A Web Content Filtering Approach. Please, be rest assured that the information you provide will be used for same purposes, and will be treated with absolute confidence and confidentiality.

Contact Information: If you have any questions or concerns about this survey or its administration, please contact egentig@gmail.com OR call/whatsapp: +234 8028411464
Thank you for your participation.

Egenti, G.E.

APPENDIX C

Questionnaire Disclaimer

Introduction: Thank you for your interest in participating in this survey. Your objective feedback is valuable and will contribute to our research and understanding of the research topic, '**Development of an Improved Web Browser Extension Filter for Online Security**'. Please take a moment to read the following information before proceeding with the Questionnaire.

Purpose: This Questionnaire is being conducted for the purpose of 'Appraising the security and control measures on social network users', 'Determining the effectiveness of the Web Content filters', 'Identifying the major limitations of web contents filtering approaches'

Confidentiality: Please be assured that your responses to this questionnaire items will remain strictly confidential. No personally identifiable information (PII) will be collected unless explicitly requested in the Questionnaire. Data collected will be used solely for the purposes outlined in this survey. Responses will be aggregated and reported in a way that ensures anonymity.

Voluntary Participation: Your participation in this survey is entirely voluntary. You are under no obligation to answer any question, and you may choose to exit the survey at any time. In the case that you choose to exit the survey before completing it, your partial responses will not be recorded or used by us.

Data Security: We are committed to ensuring the security of your data. All responses are stored securely and protected against unauthorized access. Data will be retained for the duration necessary to fulfill the survey's objectives and will be securely disposed of when no longer needed.

Data Use: The information collected through this survey will be used solely for the purposes stated in the introduction. It will not be shared or sold to third parties for marketing or commercial purposes.

Informed Consent: By proceeding with this survey, you acknowledge that you have read and understood this disclaimer and that you consent to participate voluntarily. You also confirm that you are of legal age to participate in this survey.

Contact Information: If you have any questions or concerns about this survey or its administration, please contact egentig@gmail.com OR call/whatsapp: +234 8028411464

Thank you for your participation.
Grace Egenti

QUESTIONNAIRE FOR EGENTI-FILTER SECURING ONLINE ACTIVITIES (EGENTI-FILTER SOA SURVEY)

Introduction

Thank you for participating in this survey. Your expertise in cyber security is highly valued, and your insights will contribute to a better understanding of internet filtering practices.
Please answer the following questions to the best of your knowledge and experience.

Section 1: **Demographics**

1.1. **Gender**

- Male
- Female

1.2. Age Range

- Under 18
- 18-24
- 25-34
- 35-44
- 45-54
- 55-64
- 65 and Above

1.3 Region

- Africa
- Antarctica
- Asia
- Europe
- North America
- South America
- Australia (sometimes considered a separate continent)

1.4 Years of experience in Cyber Security:

- 5-10
- 10-15
- 15 Above

1.5. Job Title/Role:
1.6. Academic Qualification
1.7. Certifications

Section 2: Awareness of Internet Filtering

2.1. Are you aware of the internet filtering measures in place within your organization's network?

- Yes
- No

2.2 Purpose of Internet Filtering:

Do you understand the primary goals and objectives of internet filtering in the context of cybersecurity?

- Yes
- No

2.3 Training and Education:

Have you received training or education on how internet filtering contributes to cybersecurity within your organization?

- Yes
- No

2.4 Perceived Importance:

How important do you believe internet filtering is for enhancing cybersecurity within your organization?

- Not Important
- Somewhat Important
- Important
- Very Important

2.5 Malware and Threat Prevention:

Do you think internet filtering helps prevent malware infections and other cyber threats?

- Yes
- No
- Not Sure

2.6 Phishing Prevention:

Do you believe that internet filtering plays a role in preventing phishing attacks and malicious email content?

- Yes
- No
- Not Sure

2.7 Data Leakage Prevention:

Do you think internet filtering helps in preventing data leaks or unauthorized data transfers?

- Yes
- No

- Not Sure

2.8 Policy Enforcement:

Are you aware of the policies and rules enforced through internet filtering to maintain cybersecurity?

- Yes
- No
- Partially

2.9 Incident Response:

In case of a cybersecurity incident, do you know if internet filtering data is used for investigation or mitigation purposes?

- Yes
- No
- Not Sure

2.10 User Responsibility:

Do you believe that users have a role in ensuring the effectiveness of internet filtering for cybersecurity?

- Yes
- No
- Not Sure

2.11 Monitoring and Reporting:

Are you aware of the monitoring and reporting mechanisms in place to assess the impact of internet filtering on cybersecurity?

- Yes
- No
- Partially

2.12 Feedback and Improvement:

Does your organization actively seek feedback from users or administrators to improve internet filtering as part of its cybersecurity strategy?

- Yes
- No

Additional Comments

Please provide any additional comments or insights related to internet filtering and cybersecurity that you deem important.----------
---

Section 3: Challenges and Concerns of Internet Filters

3.1Awareness of Internet Filtering Challenges:

Are you aware of any challenges or concerns related to internet filtering within your organization or network?

- Yes
- No

3.1 Collaboration with Other Security Measures:

How do you perceive the integration of internet filtering with other cybersecurity measures, such as firewall protection, intrusion detection, and antivirus solutions?

- Very Well Integrated
- Somewhat Integrated
- Not Well Integrated

3.3 Access to Legitimate Resources:

Have you experienced difficulty accessing legitimate resources or websites due to internet filtering?

- Yes
- No

3.4 Performance Impact:

Do you notice any slowdown in internet speed or network performance when the internet filtering system is active?

- Yes
- No

3.5 Lack of Transparency:

Do you feel that there is a lack of transparency regarding the criteria and rules used for internet filtering?

- Yes
- No

Section 4: Internet filtering Tools and Technologies

4.1 Are you aware of the presence of internet filtering tools or technologies in your organization or network?

- Yes
- No

4.2 Usage of Internet Filtering Tools:

Are you currently using internet filtering tools or technologies in your organization or on your network?

- Yes
- No

4.3 Can you list common methods or technologies used for internet filtering

- Proxy Servers
- DNS Filtering
- Firewall
- Content Filtering Software
- Application Layer
- URL filtering
- Keyword-based Filtering
- Whitelisting and Blacklisting
- Machine Learning and AI
- Other (please specify): _____

4.4 What kind of content should be blocked or restricted by web content filtering (Check all that apply)

- Adult content
- Hate speech and discriminatory content
- Malware and phishing websites
- Social media websites
- Gambling websites
- Streaming media (e.g., video and music streaming)
- Torrent and file-sharing websites
- Online shopping and auction websites
- Personal email services
- Online gaming websites
- Educational and research resources
- News and information websites
- Other (please specify): _____

Section 5: Limitations of Web Content Filters

5.1 Awareness of False Positives:

Are you aware of what false positives are in the context of internet filtering?

- Yes
- No

5.2 Personal Experience:

Have you ever encountered situations where legitimate websites or content were incorrectly blocked by the internet filtering system (false positives)?

- Yes
- No

5.3 Frequency of False Positives:

How often do you believe false positives occur in the internet filtering system?

- Rarely
- Occasionally
- Frequently

5.4 Impact of False Positives:

When false positives occur, how would you describe their impact on your work or internet usage?

- Negligible
- Mild
- Moderate
- Severe

5.5 Reporting False Positives:

Is there a mechanism in place for users to report false positives they encounter?

- Yes
- No

5.6 User Feedback on False Positives:

Do you believe that user feedback regarding false positives is actively considered and used to improve the internet filtering system?

- Yes
- No
- Not sure

5.7 Resolution of False Positives:

How satisfied are you with the resolution process when reporting false positives?

- Very Dissatisfied
- Dissatisfied
- Neutral
- Satisfied
- Very Satisfied

5.8 Awareness of False Negatives:

Are you aware of what false negatives are in the context of internet filtering?

- Yes
- No

5.9 Personal Experience:

Have you ever encountered situations where inappropriate or harmful content was not blocked by the internet filtering system (false negatives)?

- Yes
- No

5.10 Frequency of False Negatives:

How often do you believe false negatives occur in the internet filtering system?

- Rarely
- Occasionally
- Frequently

5.11 Impact of False Negatives:

When false negatives occur, how would you describe their impact on your work or internet usage?

- Negligible
- Mild
- Moderate
- Severe

5.12 Reporting False Negatives:

Is there a mechanism in place for users to report false negatives they encounter?

- Yes
- No

5.13 User Feedback on False Negatives:

Do you believe that user feedback regarding false negatives is actively considered and used to improve the internet filtering system?

- Yes
- No
- Not Sure

5.14 Resolution of False Negatives:

How satisfied are you with the resolution process when reporting false negatives?

- Very Dissatisfied
- Dissatisfied
- Neutral
- Satisfied
- Very Satisfied

5.15 Blocking Rate:

What percentage of inappropriate or unwanted content do you estimate is successfully blocked by your filtering measures?

5.16 Bypassing Attempts:

How often do you detect attempts by users to bypass the filtering measures?

- Rarely
- Occasionally
- Frequently

Section 6: Overall Effectiveness of web content filters

6.1 Effectiveness

How would you rate the overall effectiveness of your internet filtering measures on a scale of 1 to 5, with 1 being highly ineffective and 5 being highly effective?

- 1 Not effective At all
- 2 Not very effective
- 3 Neural
- 4 Somewhat Effective
- 5 Highly Effective

6.2 Filtering Goals:

What are the primary goals or objectives you aim to achieve through internet filtering in your organization?

- Protecting network security and integrity
- Preventing malware and cyber attacks
- Ensuring compliance with legal and regulatory requirements
- Enhancing productivity and minimizing distractions
- Protecting against inappropriate or offensive content

6.3 Network Security:

To what extent does internet filtering contribute to enhancing network security and protecting against cyber threats?

- Not Significant
- Somewhat Significant
- Highly Significant

6.4 Malware Prevention:

How important is internet filtering in preventing malware infections and phishing attacks?

- Not Important
- Somewhat Important
- Important

- Very Important

6.5 Legal Compliance:

Does internet filtering play a role in ensuring that your organization or institution complies with legal and regulatory requirements regarding internet usage?

- Yes
- No
- Not Sure

6.6 Productivity Enhancement:

In what ways does internet filtering help enhance productivity and reduce distractions for employees or users?

- Blocking access to social media and entertainment websites
- Restricting access to non-work-related content during work hours
- Filtering out time-wasting websites and apps
- None of the above
- Not Applicable

6.7 Content Filtering:

How important is content filtering in protecting against access to inappropriate or offensive content?

- Not Important
- Somewhat Important
- Important
- Very Important

6.8 User Education:

Does your organization provide education or guidelines to users on the purpose and goals of internet filtering?

- Yes
- No

6.9 Effectiveness Assessment:

How often does your organization assess the effectiveness of internet filtering in achieving its goals and objectives?

- Regularly
- Occasionally
- Rarely
- Not at All

6.10 Feedback Mechanism:

Is there a mechanism for users or administrators to provide feedback on the effectiveness of internet filtering in achieving its goals?

- Yes
- No

6.11 User Experience:

How satisfied are your users with the internet filtering measures in place?

- Very Dissatisfied
- Dissatisfied
- Neutral
- Satisfied
- Very Satisfied

6.12. Incident Reporting:

Do you have a system in place for users to report false positives or negatives in the filtering system?

- Yes
- No

6.13. How frequently do users report such incidents?

- Regularly
- Occasionally
- Rarely
- Not at All

6.14. Usage Metrics:

Are you tracking and analyzing user internet activity and behaviour to assess the effectiveness of filtering?

- Yes
- No

Section 7 Awareness of Data Collection:

7.1 Are you aware of what key metrics or data points are collected through the internet filtering system in your organization?

- Yes
- No

7.2 Types of Data Collected:

Please specify the types of data or metrics that are collected through the internet filtering system

- Website visit logs
- User activity
- Content categories blocked
- Other (please specify): _____

7.3 Purpose of Data Collection:

What is the primary purpose or use of the data collected through internet filtering in your organization?

- Assessing compliance with usage policies
- Monitoring user activity
- Identifying security threats and attacks
- Reporting and audit purposes
- Analyzing traffic patterns and trends
- Other (please specify): _____

7.5 Frequency of Data Collection:

How often is data collected through the internet filtering system?

- Real-time
- Daily
- Weekly
- Monthly

7.6 Data Retention Policies:

Are there defined policies and guidelines regarding data retention, archival, and deletion related to internet filtering?

- Yes
- No

Section 8: Appraising the security and control measures

8.1 Awareness of Online Security:

How aware are you of the importance of online security and privacy when using social networking platforms?

- Very Aware
- Somewhat Aware
- Not Very Aware
- Not Aware at All

8.2 Regularity of Social Media Use:

How frequently do you use social networking platforms?

- Daily
- Weekly
- Monthly
- Rarely
- Never

8.3 Privacy Settings:

Do you regularly review and update the privacy settings on your social media accounts to control who can access your information?

- Yes

- No
- Occasionally

8.4 Privacy Concerns

Please indicate your level of concern regarding the following aspects of social media use (on a scale of 1 to 5, with 1 being not concerned and 5 being extremely concerned):

- Unauthorized access to personal information
- Data breaches or leaks
- Identity theft
- Online harassment or cyberbullying
- Exposure to harmful or inappropriate content

8.5 Experiences with Privacy Incidents:

Have you ever experienced a privacy incident on social media, such as unauthorized access or exposure of personal information?

- Yes
- No

8.6 Password Security:

How strong are the passwords you use for your social media accounts?

- Very Strong
- Strong
- Moderate
- Weak
- Very Weak

8.7 Two-Factor Authentication (2FA):

Do you enable two-factor authentication (2FA) for your social media accounts to enhance security?

- Yes
- No
- Not Sure

8.8 Accepting Friend/Follower Requests:

How cautious are you when accepting friend or follower requests from unknown individuals on social media?

- Very Cautious
- Cautious
- Somewhat Cautious
- Not Cautious
- Not Applicable

8.9 Education on Online Security:

Have you ever received formal or informal education or training on online security and privacy?

- Yes
- No

8.10 Social Media Platform Information:

Do you feel adequately informed about the security and privacy features available on the social media platforms you use?

- Yes
- No
- Somewhat

8.11 Reporting Security Incidents:

Are you aware of how to report security incidents or privacy violations on social media platforms?

- Yes
- No

8.12 Supports from Social Media Companies:

How satisfied are you with the support provided by social media companies when you encounter security or privacy issues?

- Very Satisfied
- Satisfied
- Neutral
- Dissatisfied
- Very Dissatisfied

Improvement Suggestions:

1. Are there any plans or initiatives to enhance internet filtering to better achieve the organization's goals in the near future?

   _____

   _____

2. Is there anything else you would like to share regarding how your organization ensures that internet filtering does not excessively restrict legitimate access to websites or content?

   ---

3. Do you have any suggestions or recommendations for improving the internet filtering system to address challenges and concerns?

   _____

   _____

4. Is there anything else you would like to share regarding the collection and use of data through internet filtering in your organization?

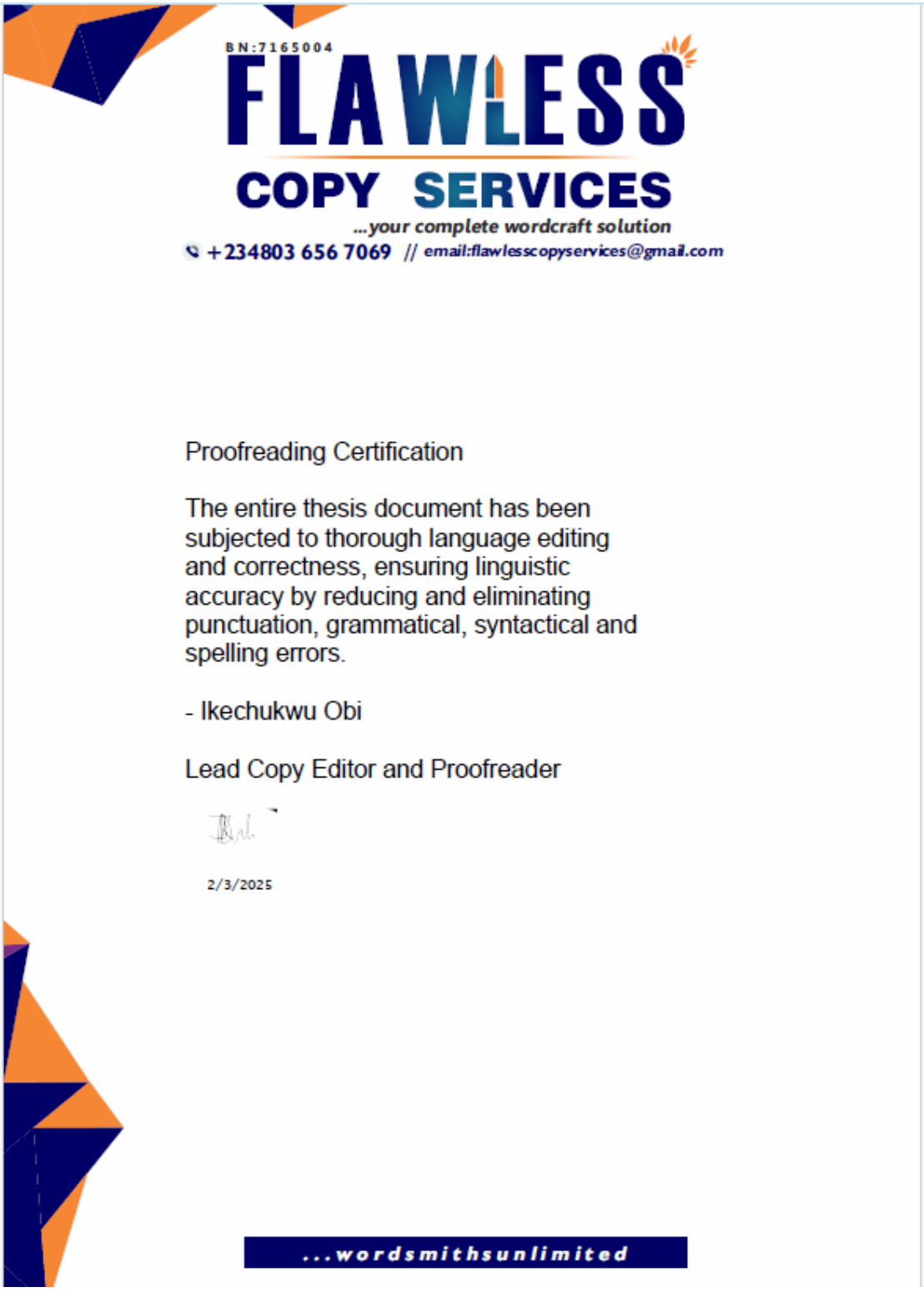
   _____

   _____

5. Are there any concerns or challenges related to internet filtering that you think may affect cybersecurity efforts negatively?

APPENDIX D

PROOFING CERTIFICATE



APPENDIX E

List of URLs in the Database that could not be accessed by the Browser Extension (for research purposes ONLY)

S/N	Links	Name/Description
1	https://banatlebanon.com	Monk4D: A website dedicated to showcasing the cultural and historical aspects of the Banat region in Lebanon. It features articles, photos, and community events
2	https://banco-bb.com	Banco do Brasil: An online banking platform offering various financial services, including personal and business banking solutions
3	https://bancodobrasil1.com	Banco do Brasil: A Brazilian bank's website that provides information on banking products, services, and online account management
4	https://cynosurejobs.net	Cynosure Corporate Solutions: The website is the online platform that offers services such as staffing and recruitment
5	https://bursasporlu.org	Bursaspor: This is a fan platform dedicated to a prominent Turkish sports club based in Bursa. This site serves as a hub for supporters, providing news, updates, and community engagement related to the club.
6	https://awaelschool.com	Awael school: This website appears to be the official website of an educational institution located somewhere in the Middle East. It provides primary and secondary education for both students, following the Saudi Arabian curriculum.
7	https://a-we.com	A-We: The website appears to be owned by a company that specializes in providing web development and digital marketing services, such as website design, and online advertising to help businesses establish and grow their online presence.

9	https:// abschlepp-taxi24.at	abschlepp-taxi24: The website provides services related to towing and vehicle transport.
10	https://bangtochat.info	ChatsForum: This website appears to be an online chat platform or forum which allows users to engage in real-time chats, share messages, and potentially connect with others based on shared interests or topics
11	https:// bandenland.be	The website specializes in the sale of tyres and related services, such as sales, installation and maintenance.
12	https://banka-navibor.by	This is for a Belarusian estate called "Банька на выбор" (translated as Bathhouse of Choice). It offers a serene environment for family or friends to unwind and enjoy various recreational activities
13	https://banketcentr.ru	The website appears to be related to banquet and event planning services in Russia. Services such as Banquet Hall Rentals, Catering Services, Event Planning, Audio/Visual Equipment.
14	https://batavia-restaurant.nl	Restaurant: This site is for Restaurant Batavia, located in Arnhem, Netherlands. This restaurant specializes in traditional Indonesian cuisine, prepared according to authentic recipes from the Indonesian culinary tradition.
15	https://batchmiami.com	Restaurant: The website likely provides details about the restaurant's location, menu, the hours of operation, and event details. Features such as online reservations, photos showcasing the restaurant's ambiance and dishes, and information about private dining and catering services are also provided.
16	https://bathboating.co.uk	Bath Boating: This site is the official website of the Bath Boating Station, a popular attraction located in Bath, UK that offers boating activities on the River Avon.
17	https://barbadostoday.bb	Barbados Today: This is an online platform for a leading multimedia news resource in Barbados. These are some of the features they offer: News Coverage,

		Multimedia Content, and Community Engagement.
18	https://batiment-metallique.net	This website focuses on metallic buildings and construction solutions. It likely provides information related to the design, manufacturing, and installation of metal structures.
19	https://batonrougekravmaga.com	This is a website related to online slot games and demos
20	https://batutatravel.com	Batuta Travel: It is a travel agency that specializes in providing travel services and experiences. It offers various options for travelers, including travel packages, tours, and destination information. The site aims to help users plan and book their trips effectively.

APPENDIX F

Summary of the performance of Egenti-Filter with related works

Author/ Year	**Mathews and Chimalakonda, (2021)**	**Norma, (2021)**	**Xuan and Nguyen, (2020)**	**This Project (Egenti-Filter - 2024)**
Aim of study	• To improve the mental health of individuals by offering the ability to control what the user views on the Internet.	•To evaluate and classify a website's features by pre-processing and entering the URL in a previously trained deep learning model	•To detect malicious websites using machine learning techniques based on URL behaviour and characteristics.	• To develop an effective real-time web content filtering model leveraging the web browser extension approach for an improved and secure users' online experience.
Algorithms Used	•Keyword-based filtering •Natural Language processing (NLP) •Multinomial Naive Bayes Classifier	Deep Learning algorithms	• Support vector machine (SVM) • Random forest (RF)	• Support vector machine (SVM) • Decision Tree (DT), and • Random Forest (RF)
Solution	• A Google Chrome extension, called the Detox Browser was designed • A popup warning is displayed when the content on any website is blacklisted	• A system was developed that could be plugged into a network's firewall or Google Chrome extension	• The researchers claimed that the proposed system is an optimized and user friendly solution for detecting malicious links.	• A web browser extension was developed to secure the activities of online users
Result	• The model	•Accuracy for the detection	• Accuracy: 93.39	• Accuracy: 83.83%

	presents the domain name of the original content to help the user judge if such content is fit for viewing. • It enables the user to totally remove any topic which is deemed fit for removal.	of malicious websites recorded 99.75% with 2.61% false hits.	• Precision: 94.67% • Recall: 92.5%	• Precision: 80.32% • Recall: 96.40%
End-user result	• The customisation of browsing experience based on personal requirements.	• The crafting of an interface where the user inserts the suspicious URL and returns the prediction output. •The assurance of a safe browsing experience by blocking non-desired content	• The result of this study was used for the crafting of a free tool for detecting malicious URLs on the Internet.	• A graphical user interface is created where the database is automatically updated • An index page that displays URL restrictions upon visiting a blacklisted link.
Limitations	• Resource Intensive • Quickly produces a lot of false positives • Only classifies English sites and does not support local languages, but only Indian headlines; as a result, the	• The Neural Network (NN) was not trained to update automatically, which shows that the NN could not be improved with every search and does not allow the addition of new feature to the model.	•No comprehensive analysis of the effectiveness of the proposed system in real-world scenarios was provided. •No detailed analysis of the computational complexity and resource requirements of the	• More dataset is needed for an improved model accuracy and performance

	experiment is region-specific.		proposed system.	
Evaluation Performance metrics	• The proposed tool is pending evaluation	• Precision • Recall • Accuracy • f1-score	•The empirical result shows the effectiveness of the proposed extracted attributes.	• Precision • Recall • Accuracy • f1-score

APPENDIX G

List of offensive words and terms

S/N	Blacklisted Keywords	Corresponding Effects
1	Swear words	Swearing is prohibited, so when someone uses it, it is often taken as a surprise, and the swearer could be thought of as rude and antisocial. Hence, swearing may have a damaging effect on how others view the swearer, which may result in misery and social isolation.
2	Offensive language in gaming	According to research, using offensive language makes people act more violently. The fouler language is used, the more heated the game becomes. Other online users, especially the young, find it uncomfortable and difficult to comprehend the context when they hear inappropriate language in games. Terms such as ass, shit, crap, fuck, arse, and so on, are some of the offensive languages.
3	Bullying	Any form of bullying can do injury to a child, both physically and mentally. Social networking users who are targeted may experience a range of issues, such as behavioural issues, anxiety, fear, depression, low self-esteem, and academic challenges. Cyber-bullying, however, may be quite destructive.
4	Drug-related promotion and sale of drugs, including drug slang	Given that the World Health Organisation has discovered that more than half of the medications from illegal Internet pharmacies are fake; illegal Internet pharmacies have begun to use social media to attract customers for their websites. This could expose large audiences, especially young ones, to potentially dangerous products.
5	Encouragement for self-harm, suicide, or eating disorders	Photos and videos that offer advice on self-harm and suicide, as well as material that openly support it and discuss techniques are accessed. Including diet

		and weight loss advice that advocates for risky and extreme measures that could endanger someone's health. False health experts' advice on medical issues that is damaging, and inaccurate information that promotes self-diagnosis can also be accessed.
6	Hate Speech on colour and creed	Due to the behaviour social networkers encounter online, young black and white kids used to detest one another and believe in their own superiority and inferiority.
7	Aggressive Language on Religion	Today's young children are becoming interested in the discussion of religious intolerance since it is on the rise. People are condemned for their religious beliefs.
8	Body Shaming and Slut-Shaming	On the internet, body shaming is a common practice. Young users, especially females frequently endure taunts such as vagina fat, tall, slender, skinny, and other derogatory terms. They also bully one another based on their outward appearances and engage in body shaming.
9	Sexual Preferences, Like LGQBT	Today's online users face numerous issues related to their sexual preferences, including issues with being gay, lesbian, queer, bisexual, or transgender. These are the ones who experience bullying most frequently due to their sexual orientations, both online and in person.
10	Violence and Extremism	Extremism and violence have affected the entire world. Extremism used to be prevalent in videos, websites, and other media. Words like young radicalization, no black should live in the United States, and many other phrases that encourage violence and extremism.
11	Sexually Explicit Phrases	These days, most online users, especially young users are smart, and are able to avoid their parents

		while talking on their smart phones, computers, and other electronic gadgets. Sexually explicit terminology like, be my emergency call, one-night stand, etc., should not be on any child's phone.
12	Sexting and Dating Terms	Online users, especially children covertly discuss sex by using phrases from sexting, such as 53x. MOS stands for mom over the shoulder, among others.
13	Virtual Violence on Video Games	There are countless video games that encourage users to engage in virtual violence. It is associated with fighting with weapons, other fights, and sexual violence video games, but the violence could be different. In addition to capturing and recording keywords connected to virtual violence in video games, parents should be aware of the video games that are causing children to become violent.

APPENDIX H

Title of Paper	Authors	Journal/Conference	ISSN	Link
Investigating the level of Information Security Awareness amongst Nigerian Tertiary Institutions	Grace Egenti, Morufu Olalere, Vivian Nwaocha, Olatunji Okesola	Third International Conference on Advances In Physical Sciences and Materials: ICAPSM 2022 18–19 August 2022 Coimbatore, India *AIP Conf. Proc.* 2901, 060025 (2023) **31** PDF Downloads	e-ISSN: 1551-7616	https://doi.org/10.1063/5.0180372
Opportunities and Barriers to Implementing Cyber-solutions in Higher Education Institutions	Grace Egenti, Olubunmi Okesola, FaladeAdesola, Oluranti Sangodoyin, Grace Jokthan, Olatunji Okesola	Journal of Information Systems Engineering and Management (JISEM)	e-ISSN:2468-4376	http://dx.doi.org/10.52783/jisem.v10i17s.2723
A Holistic Analysis of Algorithms and Approaches of Violence Crime Prediction among Students in Institutions of Learning	Falade Adesola, Grace Egenti, Olubunmi Okesola, Oluranti Sangodoyin, Grace Jokthan, Olatunji Okesola	Journal of Information Systems Engineering and Management (JISEM)	e-ISSN:2468-4376	https://doi.org/10.52783/jisem.v10i17s.2724

ENHANCED CONVOLUTIONAL NEURAL NETWORK FOR CLASSIFICATION OF MALWARES (E-CNN)

BY

KETEBU EBIEKINEN KENNEDY

ACE21140006

A Ph.D DISSERTATION SUBMITTED TO THE SCHOOL OF AFRICA CENTER ON EXCELLENCE OF TECHNOLOGICAL ENHANCED LEARNING, NATIONAL OPEN UNIVERSITY OF NIGERIA (NOUN) IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF DOCTOR OF PHILOSOPHY IN ARTIFICIAL INTELLIGENCE (Ph.D)

DECEMBER, 2023

ENHANCED CONVOLUTIONAL NEURAL NETWORK FOR CLASSIFICATION OF MALWARES (E-CNN)

BY

KETEBU EBIEKINEN KENNEDY

ACE21140006

A Ph.D DISSERTATION SUBMITTED TO THE SCHOOL OF AFRICA CENTER ON EXCELLENCE OF TECHNOLOGICAL ENHANCED LEARNING, NATIONAL OPEN UNIVERSITY OF NIGERIA (NOUN) IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF DOCTOR OF PHILOSOPHY IN ARTIFICIAL INTELLIGENCE (Ph.D)

DECEMBER, 2023

DECLARATION

I, **KETEBU EBIEKINEN KENNEDY**, hereby declare that this **Ph.D dissertation** titled Enhanced Convolutional Neural Network for Classification of Malwares has been carried out by me under the supervision Prof. Gregory O. Onwodi and Dr. Kingsley .E Ukhurebor. It has not been presented for award of any degree in any institution. All sources of information are specifically acknowledged by means of reference.

1/12/2023	
Date	Signature

CERTIFICATION PAGE

The research dissertation titled "**Enhanced Convolutional Neural Network for Classification of Malwares (E-CNN)**" meets the requirements governing the award of the Doctor of Philosophy (Ph.D) in Artificial Intelligence and is approved for its contribution to knowledge and literary representation.

DR. GREGORY. O ONWODI	13/12/2023	
Project Supervisor 1	Date	Signature

	09/12/2023	
DR. KINGSLEY. E UKHUREBOR	----------	----------
Project Supervisor 2	Date	Signature

DEDICATION

This research work is dedicated to God Almighty for His guidance and unending love towards me.

My Dad, Late Chief J.T Ketebu, My Mother, Mrs. Theresa N. Ketebu thank you for all your sacrifices in my life. God bless you all. Amen

ACKNOWLEDGMENT

This thesis was accomplished by the assistance of many individuals whom I wish to acknowledge. To my boss Isah J. Abdullahi for his support and guidance, my colleagues Edoh A. Okechalu, Joshua Mamza and Isiyaku B. Boyi

To my supervisors, Dr. Gregory .O Onwodi and Dr. Kingsley .E Ukhurebor thank you for reviewing this work, your understanding, time and energy spent in seeing me through to the final submission of this research work, may God Almighty bless you and your Family.

I want to thank Sydney, Vassal, Monique, Reagan, Comforter and my lovely wife Grace for encouraging and praying with me. Thank you so much!

May God's grace and blessings increase in all aspects of your lives, in Jesus Christ name Amen!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

In our contemporary society, the widespread use of computer systems has become integral to daily life. However, this increased reliance on technology has also given rise to a surge in cyberattacks, threatening the integrity and security of our computer systems and networks. Malware, in particular, poses a constant and serious threat to people and organizations especially the ones connected to the internet. Various sectors such as universities, schools, hospitals, manufacturing, and healthcare providers, which depend on their information systems to support critical organizational and societal functions are constantly at risk or threat from malware attacks. The escalating frequency and sophistication of malware attacks, driven by the use of automation tools in malware creation, demand continuous efforts to develop efficient and effective means of detecting and classifying malware. Researchers have explored various techniques to counter these threats, with a growing focus on converting malware samples into images, a concept known as malware visualization. This research centers on the application of Convolutional Neural Networks (CNNs) for visual detection and classification of malware. E-CNN, our proposed malware classification model, is developed using a recent malware Malevis dataset. This is critical to address the ever-changing nature of malware, empowering the model to detect new and previously unseen threats. The E-CNN model was evaluated on the widely recognized Malimg dataset, achieving an impressive accuracy score of 98.88%. Furthermore, we compared the performance of the E-CNN model to other recently cited works, also using accuracy scores to assess its effectiveness.

CHAPTER ONE

1.0 INTRODUCTION

1.1 Background of the Study

Modern day computer attacks or cyberattacks are on the increase and becoming more complex, with the increasing number of computers and mobile devices connected to the internet or cyberspace, and users accessing various digital mediums or platforms. However, this has made users, computers, and networks vulnerable to various types of attacks on the internet or in a computer environment. Malicious software, also known as malwares, are harmful programs that expose or steal sensitive data or information, as well as compromising the integrity of a computer system by preventing it from operating securely.

Malware attacks are a major source of concern for individuals and cybersecurity experts, as they have resulted in denial of services, loss of privacy, intellectual property and financial losses for victims and organizations all over the world. Machine learning techniques have been effective in malware detection and classification; however, attackers attempt to disguise malwares as legitimate files using techniques such as packing, encryption, and polymorphism; this may result in the pre-trained model making incorrect predictions (Agrawal & Khan, 2021). Recent malware attacks have become more sophisticated as a result of the use of machine learning; it is estimated that at least 230,000 malware samples are produced every day, and 18 million websites are infected with malwares each week (Ghosh, 2021).

In developing effective malware detection and classification engines, there are two main methods which are namely static and dynamic analysis. Machine learning approaches has been applied in malware analysis, the two main approaches which are static and dynamic approaches differ from each other in the manner in which features are extracted (Gibert et al., 2020) statically or dynamically (runtime execution).

There are various similar variations of malwares samples, this is because malware authors reuse the previous codes only making changes so as to form or develop new malware samples (Moussas & Andreatos, 2021).

Newer methods are being developed towards malware classification and detection, one of such areas is the area of visualization, visualization techniques has been effective in enabling and understanding complex data analytics (Keahey, 2013) or structures.

One of the first researcher to use visualization was (Nataraj et al., 2011) who represented malware samples as grayscale images in order to distinguish similarities and differences among malwares samples. This showed visual similarities of malwares belonging to the same family.

Visualization can greatly aid malware classification and does not necessitate any disassembly (static analysis) or code execution (dynamic analysis) (Moussas & Andreatos, 2021). This is because performing feature extractions of malware samples requires a degree of expert domain knowledge when performing either static or dynamic analysis.

However, Researchers have found that deep learning can produce better performance when compared to existing machine learning approaches or methods (Kalash et al., 2018). Deep learning computational approach has been successful in solving complex computational tasks, this is because of its ability to learn massive amounts of data thereby outperforming other machine learning techniques in different domains such as bioinformatics, language processing, cybersecurity, robotics, control systems and many others (Alzubaidi et al., 2021).

One of the main advantages of deep learning is its ability to execute feature engineering on its own by scanning the dataset for features that correlate with each other so as to enable faster learning.

Due to the increase rate of malware attacks, it has become very critical in how malwares are quickly identified and classified. Traditional machine learning methods are limited by feature engineering and size of data being processed, hence Deep learning has become an effective solution (Yuan, Wang, Liu, Guo, Wu, Bao, et al., 2020).

Recently, researchers have begun the use of deep learning models towards classification of malware (Cakir & Dogdu, 2018) which also has a higher predictive accuracy. Hence deep learning using neural network model is being applied towards classification of malwares. At the current rate deep learning neural network has grown to the state to surpass the limitation of machine learning techniques, this is because of the possibility of using deep learning to develop models with significantly higher number of diverse layers (Kolosnjaji et al., 2016).

1.2 NEURAL NETWORK

Neural networks are a set of algorithms which are modelled after the human brain to recognize patterns and relationship data. Neural network is also called artificial neural network (ANN) in artificial intelligence. Artificial neural network consists of artificial neurons or nodes

interconnected together with each neuron able to receive input signal, process them and send an output signal (Zhang, 2018). ANN are made up of three basic components input, hidden and output layers. The units of a layer relates with other units in the layers by connection with each connection having different numerical weight or strength (Abiodun et al., 2018). The operation of neural network is divided into two main stages namely, training and inference stage. In the training stage, the network learns from a given labelled dataset by adjusting or varying the weights of its connections so as to minimize or reduce the difference between predicted outputs and expected outputs. The network can also undergo a process called backpropagation in which the network calculates the error gradient and then updates the weights using optimizations algorithms, like gradient descent. The second stage which is the inference stage, involves using the already trained model to make predictions on new unseen data. The input data is fed into the input layer which then propagates through the hidden layers and produces and output at the output layer. Neural networks can be applied to various task such as speech recognition, image classification, natural language processing, object detection etc.

The ability of neural networks to automatically learn and extract complex patterns from large datasets has led to their phenomenal success in a variety of fields. They can handle problems that were previously challenging or impossible to solve using conventional programming techniques. Image and speech recognition, autonomous vehicles, medical diagnosis, recommendation systems, and natural language processing are a few notable applications of neural networks.

1.3 DEEP LEARNING

Deep learning is an artificial neural network (ANN) which consists of complex multilayers (Albawi et al., 2017). Deep learning is a sub field of artificial intelligence that enables computer to learn and understand complicated concepts by building hierarchy layers.(Goodfellow et al., 2016). Deep learning accepts inputs and the inputs are then passed through multiple layers which learns and extract complex patterns and representations from data. Deep learning are also called deep neural networks (DNN).

The term "deep" in deep learning refers to the depth of the neural network, which means it has multiple layers of artificial neurons. These networks are often referred to as DNNs or deep learning models. Traditional neural networks typically have only a few layers, whereas deep learning models can have dozens or even hundreds of layers, enabling them to learn hierarchical representations of data (LeCun et al., 2015).

The key advantage of deep learning is its ability to automatically learn and discover features or representations directly from raw data without the need for manual feature engineering. This is in contrast to traditional machine learning approaches that rely on handcrafted features. Deep learning models are capable of learning hierarchical representations by progressively extracting more abstract and complex features at each layer (Goodfellow et al., 2016).

Deep learning models are primarily trained using a technique called backpropagation, which involves iteratively adjusting the weights of the connections between neurons to minimize the difference between the predicted outputs and the desired outputs.

DNN have shown to be very good at complicated machine learning tasks like image classification and speech recognition. However, because of their multilayer nonlinear nature, they are opaque, making it difficult to understand how they arrive at a specific categorization or recognition (Samek et al., 2016) or learning task. Additionally, deep learning models often require a large amount of labelled training data and extensive computational resources for training, making them computationally expensive.

There are different types of deep learning architecture which are namely convolutional neural networks (CNN) for image processing, recurrent neural network (RNN) for sequential data analysis, long short-term memory network (LSTM) for language modelling and speech recognition, generative adversarial networks (GAN) for generating synthetic data Multilayer perceptron and more.

1.3.1 Convolutional Neural Network (CNN)

A CNN is a type of deep learning model specifically designed for analysing visual data, such as images and videos. The research in the area of CNNs have swiftly emerged by achieving state-of-the-art result in various computer vision tasks (Gu et al., 2018) such tasks includes image classification, object detection, segmentation, and other visual recognition tasks. CNN is a type of deep learning neural network which consist of multiple layers which is used to train dataset with very large parameters or features, this is done by converting the datapoints to images as input and combing with filters to produce a desired output (Chauhan et al., 2018). The process of training a CNN involves forward propagation and backpropagation. During forward propagation, input data is passed through the layers of the network, with the weights and biases adjusted to produce an output. The difference between the predicted output and the actual true result is measured using a loss function, such as categorical cross-entropy or mean squared error. CNN which has shown excellent performance in machine learning problems, has multiple layers

which includes convolutional layers, pooling layers, non-linearity layers and fully connected layers (Albawi et al., 2017).

CNN has two main process which are the convolution process and the sampling process. In the convolution process, the input features are applied to matrix filter in each layer to extract meaningful features, next the sampling or pooling process downsizes or compresses the feature maps while maintaining important features of the image and thereafter passed into fully connected layers.

1.4 MALWARES

Malicious software, sometimes known as malware, is damaging to computer systems because of its inherent ability to steal, damage, and interrupt computer networks and resources without the awareness of users (Tahir, 2018). Malwares are designed specifically to exploit vulnerabilities or gain unauthorized access to computer systems or networks. As a result of the rapid growth of different types of malwares, malwares were grouped into first- and second-generation malwares, while first generation malware are grouped on the basis of the manner of infection on the target system. The second generation malwares changes its structure during execution (Sahay et al., 2020).

The some of the different types of first-generation malwares are as follows: -

Worms: this is a type of malware which has the ability exist independently as a standalone program and can replicate itself across computer systems and networks thereby resulting in performance degradation (Tahir, 2018). Worms exploit security vulnerabilities to automatically propagate from one system to another, often causing network congestion and consuming system resources. Worms can also carry payloads, such as other malware or malicious activities.

Viruses: viruses affect computer systems by attaching itself to legitimate programs, executables or any file and replicates itself across a computer network, this affects the performance of the computer systems and also the network. Viruses which are self-replicating in nature causes damage by corrupting or deleting files, disrupting system operation and also stealing of sensitive information

Trojan horse: Trojan horse is a malicious code which hides it true nature of operation so as to perform a wide range of attacks on computer resources or network. Trojan horses are a common means of network attack (Yu et al., 2019). Trojan horses, are sometimes called Trojans these are deceptive programs that masquerade as legitimate software or files. Once executed, they can

perform various malicious activities, such as stealing personal information, providing unauthorized remote access to the system, or downloading and installing additional malware.

Rootkits: these are malicious modules which is loaded into the operating system (OS) kernel, which grants the module elevated privileged to perform other malicious activities such as control of the system, process hiding, information gathering and even spread of malwares.

Bots: this is a type of malware which infects computer or group of systems which enables the attacker to control the computer systems remotely from a central command and control server so as to launch cyber-attacks such as distributed denial of services DDOS, sending of spam mails or distribution of additional malwares. A network of bots-controlled computers is called botnets.

Keylogger: this is a malicious tool which is mostly installed without the knowledge or permission of the user. Keylogger saves all keystroke generated by the user through the machine so as to monitor and steal vital or sensitive information without user's consent.(Wajahat et al., 2019)

Ransomware: this is a type of malware which encrypts or locks data of a victim computer by performing a significant number of file related operation in a short period of time (Bae et al., 2020) and may be released upon payment by victim. It often spreads through phishing emails, malicious downloads, or exploit kits. Ransomware attacks have become increasingly prevalent, targeting individuals, businesses, and even critical infrastructure.

Spyware: Spyware is a type of malicious tool which keep tracks of all the user's activities performed on the computer and the information sent back to hacker or creator. Spyware is designed to covertly gather information about a user's activities and transmit it to a third party without the user's consent. It can track keystrokes, capture screenshots, monitor browsing habits, and collect sensitive data like login credentials or credit card information. Spyware is often used for surveillance, identity theft, or unauthorized advertising purposes.

Adware: Adware, short for advertising-supported software, is typically not as malicious as other forms of malware. It displays unwanted advertisements or redirects users to advertising websites, often bundled with free software downloads. While adware may be more of a nuisance, it can impact system performance and compromise user privacy.

The second generation of malware represents a significant evolution in malicious software, introducing more advanced techniques and capabilities compared to its predecessors. This generation of malware emerged in the late 1990s and early 2000s, building upon the foundation

laid by the first generation of viruses and worms. This second-generation type of malware has the ability to change or hides its structure or conceals by encrypting its true nature of operation so as to evade detection by malware detectors. They can be classified into polymorphic, oligomorphic, encrypted, blended threat malwares and metamorphic malwares.

Encrypted malwares: malware creators use encryption and decryption methods, so as to avoid detection and static code analysis. Recently, Transport layer security (TLS) protocol which is widely used in securing application data, is now being used by malware authors to encrypt malware traffic thereby making malware detection such as deep packet inspection (DPI) ineffective (Liu et al., 2019).

Oligomorphic malwares: This is comparable to encrypted malware, but it differs in that each new infection or attack requires a unique decryptor, which is chosen from a list of decryptors. This makes it difficult for anti-malware engines to detect it; nevertheless, if the anti-malware engine scans all existing decryptors, detection is still feasible.

Polymorphic malwares: The high number of distinct malware samples found each day shows that there is likely a lot of code reuse going on beneath the layers of stealth (Deng & Mirkovic, 2022). These types of malwares can appear to be unique but it functionalities is the same as other malware samples such malwares are polymorphic in nature.

Metamorphic malwares: This is a self-modifying malware that alters the structure of its code while maintaining its functioning so as to evade detection (Mumtaz et al., 2021). Metamorphic malware may or may not need decryptors to appear as a unique malware sample. This is because of its mechanism which changes its syntax after each copy, however its mode of attack or workings does not change.

Blended Threats: Second-generation malware introduced the concept of blended threats, which combined multiple attack vectors or malicious functionalities. For example, a malware program might combine a worm to spread itself, a Trojan to perform unauthorized actions, and a rootkit to hide its activities. Blended threats increased the sophistication and effectiveness of malware attacks, making them more potent and harder to combat.

Advanced persistent threats (APTs) Malwares. These type of malware are designed specifically to infiltrate and compromise specific persons or organizations using multi-steps process over a period of time (Rot & Olszewski, 2017). This advanced type of malware uses

complex tools such as zero-day exploits and social engineering so as to make its attacks more effective.

It's important to note that the field of malware is constantly evolving, with new types and variants emerging regularly. As such, staying informed about the latest threats and implementing appropriate security measures is crucial to protect against these malicious programs.

1.5 PROBLEM STATEMENT

The level of sophistication of malwares is on the increase as well as the rate of malware attacks on computer systems and networks with increase in internet use, malwares pose serious threat to the digital world and its impact severe. Malware developers or attackers are employing different techniques which makes malwares evade detection and classification hence causing serious harm to information, computers and networks. Researchers are developing new methods in detection and classification of malwares with high accuracy which has been effective. One of such technique is the use of CNN in developing models. This works attempts to develop a model which will be an improvement of the existing method CNN, by improving upon existing techniques, this research strives to contribute to the advancement of malware detection and classification by better predictive performance in terms of efficiency and accuracy.

By addressing the limitations of current approaches and leveraging the potential of CNNs, this study aims to provide a valuable contribution to the field of malware detection and classification. The outcomes of this research have the potential to significantly impact the effectiveness and efficiency of combating malware threats in the digital landscape.

1.6 AIM AND OBJECTIVES OF STUDY

The main aim of this desertion is to develop a CNN for accurate malware classification. The proposed model will be trained and feature extraction will be performed on image representations of binaries or executables, resulting in improved predictive performance and results.

The objectives of the research work are outlined as follows:

1. Develop an effective deep CNN model specifically designed for malware detection and classification. By leveraging the power of deep learning techniques, the model will be able to effectively identify and categorize different types of malwares based on their binary or executable representations.

2. Utilize existing pre-trained deep learning architectures and apply transfer learning methods to train the model for malware classification. Transfer learning allows the model to leverage knowledge gained from training on large-scale datasets and adapt it to the task of malware classification. By utilizing pre-trained architectures as a starting point, the model can benefit from the learned features and accelerate the training process.
3. Compare and analyse the results obtained from the newly developed deep CNN model with the models developed using transfer learning methods. By evaluating the performance of different models, this research aims to determine the most efficient and accurate approach for malware classification. The analysis will consider factors such as classification accuracy, computational efficiency, and overall effectiveness.
4. Applying the newly developed model on benchmark dataset and compare its performance with other researchers work to access its effectiveness and benchmark against state-of-the-art approaches.

By achieving these objectives, this research aims to contribute to the advancement of malware classification techniques. The developed deep CNN model, along with the comparison and analysis of different approaches, will provide valuable insights into the most effective methods for accurately identifying and categorizing malware.

1.7 SCOPE OF STUDY

This study focuses on developing a novel CNN model and conduct a comprehensive comparison of its results and performance with existing pre-trained deep learning CNN architectures. The focus will be on applying the models (new and pre trained architecture via transfer learning techniques) to a dataset. The recent dataset utilized in this study comprises of thousands of recent image representation of executable, encompassing both malicious or malware samples and benign programs.

By focusing on the development of a new CNN model and comparing it with an established pre-trained CNN architecture, this study seeks to advance the field of malware classification. The evaluation and comparison of these models' performance will provide valuable insights into the effectiveness and potential improvements in identifying and classifying malware. The dataset, consisting of diverse and up-to-date image representations of executables which will ensure a comprehensive evaluation of the models' capabilities in handling real-world malware samples.

1.8 SIGNIFICANCE OF THE STUDY

This research holds significant importance in the field of malware detection and classification by introducing a newly developed model based on CNN. Through the application and comparison of this model, valuable insights will be gained, aiding malware researchers in identifying the most effective convolutional neural network models for malware detection and classification.

The developed models have practical implications as they can be leveraged by cybersecurity analysts to create robust malware detection and classification tools. By incorporating these models into existing security systems, computer systems can be effectively shielded from the threats posed by malicious programs or codes. This contributes to enhancing information security and reinforces the defence mechanisms of computer networks against evolving malware attacks.

Moreover, the research outcomes have the potential to drive advancements in the field of cybersecurity. By identifying the best performing CNN models for malware detection and classification, future research and development efforts can be directed towards refining and optimizing these models, leading to more accurate and efficient malware identification techniques.

Ultimately, the significance of this study lies in its contribution to the improvement of malware detection and classification methodologies, strengthening the defence against malware and fostering a more secure digital environment.

1.9 OUTLINE OF DISSERTATION

- Chapter 1: Introduction
- Chapter 2: Literature Review
- Chapter 3: Research Methodology
- Chapter 4: Results and Analysis
- Chapter 5: Conclusion

1.10 SUMMARY OF CHAPTER

This chapter delves into the evolving threats and the severe consequences associated with malware attacks on digital data, computer systems, and networks. Despite notable achievements in malware detection and classification, malware creators continuously strive to enhance the

sophistication of their malicious software. They employ a wide range of techniques to evade detection and inflict significant damage on digital information and its associated resources.

To address this ever-evolving landscape of malware, researchers are consistently refining existing methods of malware detection and classification while also exploring new techniques. This research places particular emphasis on leveraging deep learning convolutional neural networks (CNNs) to develop an effective model capable of efficiently analysing, detecting, and classifying malware with substantially higher accuracy than previous approaches.

By employing CNNs, which have demonstrated success in various domains, the research aims to push the boundaries of malware detection and classification. The focus lies on developing a robust model that can adapt to the evolving nature of malware, enabling accurate identification and classification even in the presence of sophisticated evasion techniques.

The outcomes of this research have the potential to significantly enhance the field of malware analysis by providing an advanced model capable of tackling the ever-increasing challenges posed by malware creators. By improving the accuracy and efficiency of malware detection and classification, the research contributes to bolstering the security of digital systems and networks, safeguarding valuable information from malicious attacks.

CHAPTER TWO

2.0 LITERATURE REVIEW

This chapter provide an in-depth examination of researchers' works and techniques for malware classification using deep learning methods. In addition to exploring these approaches, it is also necessary to briefly discuss malware analysis and detection techniques. This is due to the fact that understanding how malware analysis and detection are carried out aids in understanding how well malware classification models perform.

2.1 MALWARE ANALYSIS

Malware analysis is simply a process of analysing malware samples so as to determine its method of operation (functionality, behaviour and impact) on computer systems and network. This is done by extracting information about the malware samples, the information extracted helps in understanding the nature and scope of functionality of the malware sample. Malware analysis helps in categorizing the type of malware sample, i.e. whether the sample is a botnet, virus, ransomware etc. Malware analysis is a vital process towards developing effective detectors this is because useful information (registry keys, filenames, signatures) which are extracted and studied by researchers towards improving and making better future detectors.

TYPES OF MALWARE ANALYSIS

Various techniques are employed to analyse malwares into different categories.

Malware analysis can be broadly classified into three types which are namely

1. Static analysis
2. Dynamic analysis
3. Behaviour analysis

STATIC ANALYSIS: in static analysis, the malware samples are analysed without executing or running the malware, however all necessary information about the malware is extracted. The extracted information can be used to form detection patterns. when static analysis is performed, file information such as string signature, opcode frequency, windows API, control flow graph (CFG), byte sequence n-grams are used as technical indicators for determining whether a file is malicious or not.

DYNAMIC ANALYSIS: in dynamic analysis, the malware sample is executed in a sandbox (safe and controlled) environment so as to analyse its functionality and behaviour during runtime using debugging tools. This is a method of analysis which gives malware researchers deep visibility or insights on the nature of (potential) threat or actions at runtime execution.

BEHAVIOURAL ANALYSIS: this involves analysing and interacting with the suspicious malware samples after execution. It involves the monitoring the processes, registries, memory usage, cpu usage, data transfer and other computer system resources so as to determine its method of operation of the malware sample. Behavioural analysis is a time consuming and complicated process which requires advanced skills.

2.2 MALWARE DETECTION TECHNIQUES

Malware detection refers to the process of identifying and detecting a (suspicious) file or program as malicious or benign on a system or network. Thereby preventing computer systems from incidents such as system compromise, data and information loss. Malware detection techniques can broadly be divided into three categories.

1. Signature based detection
2. Heuristic based detection
3. Specification based detection

SIGNATURE BASED DETECTION

In signature-based detection, the suspected file is disassembled into sequence of bytes which is known as a signature. This signature is then compared with an existing database of known malware signatures to determine if the file is malicious and which family of malware it belongs to. This detection method is usually used by most antivirus programs.

HEURISTICS BASED DETECTION

This is a behaviour-based method of detection in which differentiates between normal and abnormal behaviour of a system. Heuristics based detection process entails a detailed study or observation of the system in an idea condition and in absence of an attack which will be used as a baseline for comparison on the system in the event of a malware attack. This method is effective in detecting unknown malwares or new threats; however, it is a resource intensive method such as use of virtualized environment and usually prone to a high level of false positives.

SPECIFICATION BASED DETECTION

In specification-based detection, rule sets are defined which specifies the valid or intended behaviour exhibited by any program of the system. Specified based detection involves observing and monitoring programs executions so as to determine malicious activities by detecting deviations of their behaviour from previously specified rule sets. It overcomes the limitation usually faced by heuristics-based detection by reducing the level of false positive and increasing the level of false negative.

2.3 MALWARE NORMALIZATION

Malware writers and attacker use often use obfuscation techniques to hide or transform the program codes executables of the malware so as to hide malicious intent and evade detection. Hence the use of malware normalization systems which processes obfuscated program executables and eliminates the obfuscation so as to reveal the true nature of the program codes executables, this helps to improve detection rate. For malwares developed using toolkits (such as UPX, VirtTool, etc) normalization approaches can be employed to improve the detection rate of a malware detector (Dwivedi P & Sharan H).

2.4 DEEP LEARNING METHODS FOR MALWARE CLASSIFICATION

Deep learning, a subset of machine learning, has gained significant attention and shown promise in various domains, including malware classification. Deep learning models, such as CNNs and RNNs, have demonstrated remarkable capabilities in extracting intricate patterns and features from complex data, making them well-suited for malware analysis.

CNNs have been widely employed in malware classification tasks due to their ability to automatically learn hierarchical representations of data, particularly in image-based malware analysis. These models utilize convolutional layers to extract local features from images of malware binaries or executables, followed by pooling layers to capture high-level representations. The extracted features are then fed into fully connected layers for classification.

RNNs, on the other hand, are effective in capturing temporal dependencies and sequential patterns, which are valuable in analysing the behaviour and dynamic aspects of malware. These models, such as LSTM networks, can process sequences of system calls, network traffic, or other time-series data to identify malicious patterns.

2.4.1 CONVOLUTIONAL NEURAL NETWORKS (CNN)

CNN is a type of neural network which is frequently used in the field of computer vision for tasks such as image classification, object recognition and detection. CNN consists of multiple (input, hidden and output) layers of artificial neurons which processes images to identify unique patterns or feature representations. The convolutional layers learn feature representations by extracting local characteristics of from inputs or previous layers so as to obtain a new feature (Guo et al., 2017).

COMPONENTS OF CNN

CNN consist of components called layers. There are broadly three types of CNN layers, namely

1. Convolutional Layers
2. Pooling Layers
3. Fully connected layers

2.5 RELATED WORKS

This section provides a comprehensive analysis of various researchers' approaches in the field of malware classification using deep learning methods. By exploring the works of different researchers, a comprehensive understanding of the advancements and contributions in the application of deep learning for malware classification is gained.

Researchers (Meng et al., 2017) developed a model called malware classification model based on static malware gene sequences (MCSMGS), this model uses genetic theories to analyse malwares in which malware code fragments which carries functional information are referred to as malware gene sequences. The model extract API call sequences from malware gene which are converted into n by k two-dimensional matrix (where n is length of sequence and k is dimensional space) so as to represent intrinsic correlation and similarity. Thereafter CNN model is used for analysing and classification on the malware gene sequences. The result of the experiment achieves an accuracy of 98% on Microsoft challenge dataset.

(Kalash et al., 2018) proposed a deep learning framework for malware classification using deep convolutional neural network (CNN) architecture, which is referred to as M-CNN model. The model processes grayscale images of binaries from two datasets (Malimg and Microsoft malware dataset). The result of the experiments achieved an accuracy score of 98.52% and 99.7% on Malimg and Microsoft malware dataset respectively.

In a research work by (Le et al., 2018), a model was developed which combines a convolutional neural network plus two bi-directional long short term memory architectures (CNN-BiLSTM) for malware classification. A generic image scaling algorithm which interprets the malware file byte code as a one-dimensional image with a fixed target size. The generated images is fed to the CNN-BiLSTM model in which the output of convolutional layers are connected to one forward LSTM layer and one backward layer. The two outputs are then fed to the output layer of the model, the result of model achieved an average accuracy score of 98.8%.

In a research work by (Lo et al., 2019), the researchers performed malware classification using a special CNN architecture Xception model based which its experiment was based on Malimg and Microsoft malware dataset. This approach performs malware classification using two file types (.byte and .asm) in which the predictions are stacked together so as to give a predictive result. This helps to reduce overfitting problem as well as achieved a very high accuracy 99.03%. The Xception model was very effective and less time consuming when compared to other methods such as KNN, SVM and VGG16.

In a research work by (Khan et al., 2019), the researchers based their research work on two pre trained architectures which are GoogleNet and ResNet152. These architectures were applied on the Microsoft malware classification challenge dataset which contains malware binaries which are converted to images. GoogleNet was the fastest among the two models achieving an accuracy score of 74.5% while ResNet152 achieved an accuracy score of 88.36%

A model framework was proposed by researchers (Yuan, Wang, Liu, Guo, Wu, & Bao, 2020) to improve malware classification accuracy using markow images. This model was called byte level malware classification method based on markov images and deep learning (MDMC). This entails converting malware binaries into markov images using markov transfer probability matrix so as to retain global statistics of malware bytes. The generated markov (malware) images has a fixed sized which reduces redundancy of bytes information. The structure of convolutional neural network (CNN) is based on VGG16. The experiments were conducted on two datasets which are Microsoft and Drebin malware dataset. The average accuracy rates where 99.264% (Microsoft dataset) and 97.364% (Drebin dataset).

(Nisa et al., 2020) proposed a hybrid method of malware classification which involves a combination of pre-trained deep convolutional neural network model (Alexnet and Inception-

v3) and scaled feature texture analyser (SFTA) which are used for feature extraction, the results of the feature extraction are then combine into a single feature vector using serial-based feature technique, while using principal component analysis (PCA) for selection the most informative or relevant features. The result of the experiment when applied on the Malimg image dataset achieved an accuracy of 99.3%

A research work by (Bensaoud et al., 2020), the researchers selected six deep learning models for static malware classification in which three models were combine with support vector machines algorithm(SVM) to enhance the neural network models which are MLP-SVM, CNN-SVM, and GRU-SVM. The experiment was performed on the Malimg dataset which contains images of converted malware binaries. The results showed that the pre-trained architecture model Inception-V3 achieved an accuracy score of 99.24%

Researchers (Yoo et al., 2021), proposed a machine learning hybrid model called the Al-Hydra, this model combines random forest (RF) and Multi-layered perceptron (MLP) which are very effective for malware detection. This model which consists of four sub classification models (static RF, Dynamic RF, static MLP and Dynamic MLP) uses a voting scheme in which a rule-based majority vote is used to determine if a sample is malicious or benign. The results of the experiment showed Al-Hydra having an average accuracy of 85.1% using KISA dataset.

A research work by (Kumar, 2021), who developed a model using transfer learning called malware classification with fine-tune convolutional neural networks (MCFT-CNN). This model was developed by altering the last layer with a fully connected dense layer of a pre-trained existing model ResNet50. The MCFT-CNN model when trained with Malimg dataset achieve an accuracy of 99.18% and 98.63% on Microsoft malware challenge dataset.

In another study, (Awan et al., 2021) proposed a model based on deep learning framework called spatial attention and convolutional neural network (SACNN) for malware classification. This model represents a simple solution which does not require generated images from binaries to undergo special preprocesing operations such as data augmentation or feature engineering in order to solve malware classification problems. The model consists of a transfer learning model (VGG19), a dynamic spatial attention mechanism which focuses on only important areas of the generated images for malware classification. The result of experiment when applied to Malimg malware dataset produced an accuracy of 97.68%.

Researchers (Prajapati & Stamp, 2021) conducted CNN experiments in which transfer learning played a vital aspect. The pre trained models used are VGG-19 and ResNet152. The dataset used consist of 20 different malwares families and is a combination of the Malicia dataset and the Microsoft dataset. The malware samples are converted in images and the early portion or layers of the models frozen while the last few parts of the layers are retrained. The result of the experiment achieved and accuracy score of 92.16% for VGG19 while the ResNet152 was 91.50%.

(Asam et al., 2021) proposed a malware classification framework called Deep Feature Space-based malware classification (DFS-MC), the proposed model entails customizing and fine tuning ResNet-18 and DensNet-201 in combination with SVM. The hybrid model learning scheme involves extracting deep ensemble features of customized CNN models and then applying SVM classifier for malware classification on deep ensemble feature space. The proposed model produced an accuracy of 98.61%.

Researchers (Carletti et al., 2021), carried out an evaluation so as to determine the robustness of CNN for malware classification. This was done by specializing existing CNN models (ResNet50, InceptionV3, MobileNet and VGG16) on malware images through transfer learning for malware classification. In accessing the robustness of the models, the malware samples input are perturbed which involves subjecting the original executables through obfuscation methods. A metamorphic technique such as dead code insertion was applied directly on the hexadecimal representation of a binary file, this involves inserting junk codes into the text section of the binary file. The BIG2015 dataset used in the experiment with the experiment being in two folds with malware classification on the original dataset and accessing robustness on obfuscated dataset. The overall best CNN model was MobileNet which a high accuracy score of 99.25% and on obfuscated dataset 96.2% showing the CNN model is very robust for malware classification.

(Schofield et al., 2021) Presented a CNN model malware classification based on Windows system Application Program Interface (API) call. The researchers identified API call sequences as an important feature for malware classification, this is because API calls shows system calls or events on windows operating system occurring during runtime of a malicious file sample. The research work used a database of API call streams. The model uses both one-dimensional (1-D) CNN and term frequency-inverse document frequency (TF-IDF) in mapping API call streams. The result of the experiment showed the 1-D CNN model achieving an accuracy score of 98.17%.

In a recent study, researchers (Marin et al., 2022) developed a model for malware classification, the experiments consisted of two dataset which are the Malimg and the Microsoft challenge dataset. This involves the dataset being converted, processed and resized to a define size (64x64 pixels) so as to enable the model make accurate classification. Experimental results showed that model was efficient and effective with an accuracy score of 98.70% on both datasets.

Researchers (Lin & Yeh, 2022) proposed a bit and byte-level sequence one dimensional (1D) CNN model which extracts vital features from the one dimensional structure of binary executables, instead of converting executables into two dimensional images (2D) which makes it difficult to determine a fixed width with all inherited sequential structures within the byte-level sequence. Resizing and compression methods are applied to fix the length of each byte-level sequence, additionally bit transformation is applied so as to expand the byte-level to bit level sequences. This is because each machine instruction is encoded as 8 bits. The model maintains the contextual information for the machine instructions and also has fewer number of parameters in comparison to 2D CNN models. The model when applied to Microsoft malware challenge dataset achieved an accuracy score of 98.7% for malware classification.

In a research work by (O'Shaughnessy & Sheridan, 2022), one area of concern was malware developers employing obfuscation techniques so as to evade detection. Hence a hybrid framework for malware classification was developed to overcome the challenges faced by other image-based malware classification models. This framework combines the strengths of both static and dynamic analysis to overcome obfuscated malware samples. This is done by converting malware samples into two dimensional images mapped through space filled curve (SFC) traversals. This is important because the data structures of resulting SFC images of original malware samples are maintained after conversion. The result of the experiment when applied to the dataset gave an accuracy score of 97.6%.

Researcher (Alshamrani, 2022) developed a novel approach using deep learning to categorize malwares families and multi classification. This was done by converting malware samples into sequence of pixel values producing two-dimensional matrix grayscale images. The CNN model uses entropy filters to find distinct patterns in the image processed. The performance of the CNN model was evaluated using malware dataset of 10,000 samples with nine classes. The result of the model achieved and accuracy score of 99.7%.

A research work by (Onoja et al., 2022) developed a malware detection and classification model whose goal was not only to enhance effective detection of malware but also to reduce the

prediction time. This was achieved by proposing a hybrid model which integrates XceptionCNN with LightGBM algorithm. The model was applied on the Malimg dataset which contains 9339 gray scale images of malware sample of 25 different classes of malware and 1042 benign samples. The model achieved and accuracy of 99.85% for binary classification and 97.40% for multi-classification.

Researchers (Hammad et al., 2022) developed a model for malware classification, which performed best among the experiment conducted. The proposed model involved using deep feature technique (GoogleNet) to extract features from the Malimg dataset, while KNN is used for classification. This achieved the highest accuracy score of 96.64%.

In a recent study, researchers (Ahmed et al., 2023) formulated malware signatures as 2D image representation in classifying malwares using deep learning techniques on Microsoft malware challenge BIG 2015 dataset which contains malware samples. The model was developed using transfer learning of Inception V3 architecture and its performance produced a classification accuracy score of 98.76%. The research work compares its performance with various machine learning and deep learning technologies towards malware classification such as Logistic Regression (LR), Artificial Neural Network (ANN), Convolutional Neural Network (CNN), transfer learning on CNN and Long Short-Term Memory (LSTM).

Table 2.1: Table showing summary of malware classification models

S\N	Author	Classification Model	Dataset	Accuracy	Limitation
1	(Meng, Shan et al. 2017)	MCSMGS	Microsoft challenge dataset (BIG 2015)	98.0%	The dataset used is limited due to its robustness
2	(Kalash et al., 2018)	M- CNN	Malimg Microsoft challenge dataset (BIG 2015)	98.52% 99.7%	The dataset lacks robustness
3	(Le et al., 2018)	CNN-BiLSTM	Microsoft challenge dataset (BIG 2015)	98.8%	Slow training time, Imbalance dataset
4	(Lo, Yang, & Wang, 2019)	Xception	Malimg Microsoft challenge dataset (BIG 2015)	99.03% 99.97%	The dataset lacks robustness

5	(Khan, Zhang et al. 2019	GoogleNet ResNet152	Microsoft challenge dataset (BIG 2015)	74.5% 88.36%	The dataset used lacks robustness
6	(Yuan, Wang, Liu, Guo, Wu, & Bao, 2020)	MDMC model	Malimg Derbin Dataset	99.264% 97.364%	Processing time The dataset lacks robustness
7	(Bensaoud, Abudawaood et al. 2020)	Inception V3	Malimg	99.24%	Dataset imbalance, high computation time
8	Yoo, Kim, Kim, & Kang, 2021	Al-hydra	KISA Dataset	85.1%	Uses high computation to extract various features, Uses voting mechanism to decide classification, this is suspectiple to high false (FP) rate of begnin samples
9	(Prajapati & Stamp, 2021)	VGG19 ResNet152	Malicia Dataset	92.16% 91.50%	High computation time, dataset imbalance
10	(Kumar, 2021)	(MCFT-CNN).	Microsoft challenge dataset (BIG 2015) Malimg	98.63% 99.18%	Due to the size of complex architecture of resnet50 has high computation overhead
11	(Awan et al., 2021)	SACNN	Malimg	97.68%	Dataset imbalance, lack of the exploration in the data augmentation and the feature engineering domains.
12	Asam, Khan, Jamal, Zahoora, & Khan, 2021	DFS-MC	Malimg	98.61%	Very large processing and computation cost
13	Carletti, Greco, Saggese, & Vento, 2021)	MobileNet Xception XGBoost	Microsoft challenge dataset (BIG 2015)	99.25% 99.07% 99.43%	Accuracy of models drops considerable on obfuscated samples
14	(Lin and Yeh 2022	Byte-level 1D CNN	Microsoft challenge dataset (BIG 2015)	98.7%	Model did not always produce better performance while binary executables were converted and resized to larger images.
15	(O'Shaughnessy & Sheridan, 2022)	SFC KNN-HOG	VirusTotal dataset	97.6%	Long conversion time of malware samples to SFC images
16	Onoja, Jegede et al. 2022	Xception+LightGBM	Malimg	97.40%	The dataset lacks robustness, imbalanced dataset
17	Hammad, Jamil et al. 2022	GoogleNet+KNN	Malimg	96.40%	High computation time
18	(Alshamrani, 2022)	Binary code to pixel vector transformation	Microsoft challenge dataset (BIG 2015)	99.97%	The dataset lacks robustness, model susceptible to overfitting
19	Ahmed, Afreen, Ahmed, Sameer, & Ahamed, 2023	Inception V3	Microsoft challenge dataset (BIG 2015)	98.76%	The dataset lacks robustness

After a comprehensive examination of various researchers' works, it becomes clear that some of these models have been developed by adapting existing architectures through transfer learning, while others are created by combining two or more architectures or crafting entirely novel models from scratch. In Table 2.1, we provide a summarized overview of crucial information, including the dataset used, accuracy scores, and limitations of the models discussed in Section 2.5. These advancements play a pivotal role in enhancing the effectiveness and adaptability of solutions for detecting and classifying malware, ultimately strengthening the security of computer systems and networks.

2.6 RESEARCH GAPS

A notable research gap in the field of malware classification can be identified regarding the utilization of recent and evolving malware datasets in developing models. Many researchers have relied on existing works that employ datasets that may not encompass the most up-to-date malware samples. This limitation arises from the difficulty in obtaining access to publicly standardized or benchmarked datasets containing recent and emerging malware samples. However, there is a clear need to address this gap by developing models that are trained on recent malware datasets, providing a representation of the ever-evolving nature of malware threats.

The absence of up-to-date datasets poses a challenge in evaluating the performance and effectiveness of malware classification models in real-world scenarios. As malware constantly evolves and adapts to evade detection, it is crucial to train models on datasets that encompass the latest malware samples. This ensures that the models are equipped to accurately identify and classify the most recent and sophisticated malware variants.

CHAPTER THREE

3.0 RESEARCH METHODOLOGY

In other to comprehend the proposed research methodology, this chapter discusses concepts, theories and technologies as well as other vital information necessary to understand the suggested approach of malware analysis and classification on malware image dataset using a deep convolutional neural network architecture (CNN). Also, this methodology will analyse different learning rates, as well as optimizing the convolution layers, pooling layers, activation function and other optimal parameters which will be effective for developing a model classification of malware images. The methodology of this study is divided into data gathering, CNN architecture design and classification.

3.1 CONVOLUTIONAL NEURAL NETWORK (CNN)

Convolution is simply featuring transformation on given image which brings out or shows hidden patterns when passed through a filter (kernel). The filter finds or identifies patterns when applied on the image. With the discovery of high-level features, more abstract interpretation of the image dataset can be found. The use of CNN models is mainly applied in solving computer vision problems such as object tracking, object identification, image identification, feature identification etc.

CNN is among the best state of the art neural network architecture used for image classification tasks or problems. CNN consist of various components or layers which is illustrated in Figure 3.1 however, the three main components are namely convolutional layer, pooling layers and fully connected layers.

3.1.1 CONVOLUTIONAL LAYERS

Convolutional layers are usually one of the first layers and also are foundational or core building blocks of the convolutional neural network. Convolution is simply a mathematical procedure that requires two inputs, like an image matrix and a set of filters whose parameters must be learned.

These layers consist of many filters, which are also referred to as kernels (matrix) which extracts local or regional characteristics features from the input images. The filters which usually have a much smaller spatial dimension than the input images are then extended across the input images

or previous feature maps in sequence, this is referred to as a stride, thereafter and then passed through an activation function which forms a feature map.

As mentioned, this layer which consists of a set of filters or kernels produces an output as a result of a linear operation (i.e. linear multiplication of matrices weights and inputs) and then stacking of feature maps of all filters along the depth dimension of the image.

3.1.2 POOLING LAYERS

The primary objective of the pooling layer is to gradually reduce the spatial (parameter) size or dimensions of the input. Consequently, by doing so, the number of parameters to learn and the amount of computation are both reduced. This process is done by down sampling feature maps which simply involves summarizing vital features in patches or regions of the feature maps. The addition of a pooling layer after the convolutional layer is a common technique used for ordering layers within a convolutional neural network that may be performed one or more times in a given model. The pooling layer operates on each feature map separately to generate a new set of the same number of pooled feature maps. There are three main types of pooling functions which are namely max pooling, average pooling and global pooling.

3.1.2.1 MAX POOLING

This is a type of mathematical operation which selects the highest or maximum element value from the region of the feature map covered by the filter. Hence the output of the pooling layer will produce a feature map which contains the most important features of the feature map.

3.1.2.2 AVERAGE POOLING

This is a type of mathematical operation which is achieved by selecting the average element value from the region of the feature map covered by the filter. This involves calculating the average value from a portion of the image using a specific size.

3.1.2.3 GLOBAL POOLING

This is another type of pooling which is used to down sample a feature map into a single value using either global max pooling or global average pooling. In some CNN models global pooling is often used as a replacement for fully connected layer.

3.1.3 FULLY CONNECTED LAYER

This is usually the last layer of the CNN architecture which receives its input as an output from the pooling layer which is a representation of high-level features of the image, are then flattened into a vector so as to perform classification operation based on features extracted.

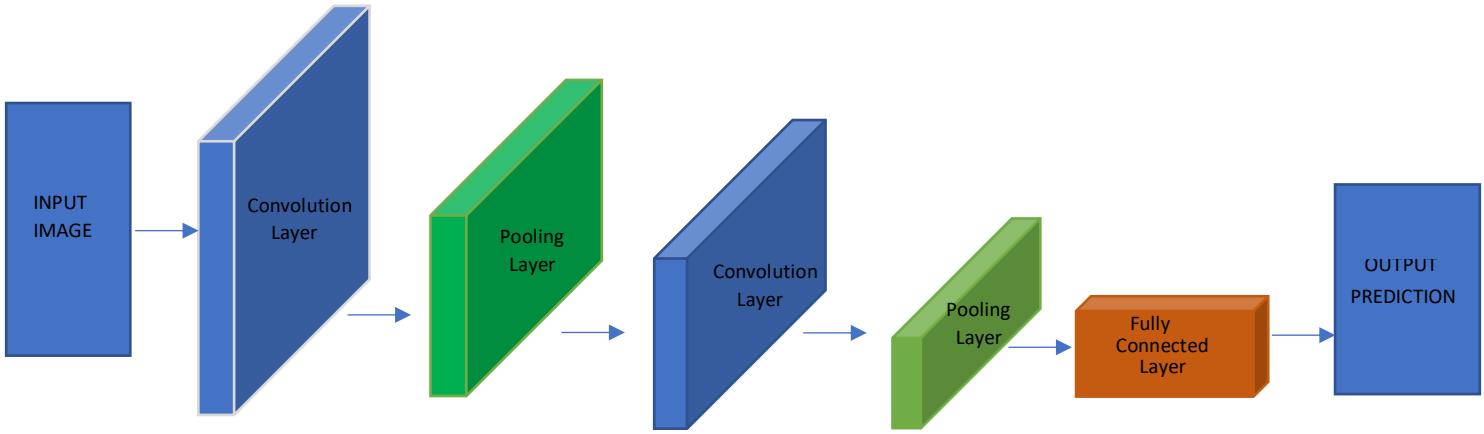


Figure 3.1: Showing basic components of CNN

3.2 TRANSFER LEARNING

This is a deep learning technique which uses an existing trained model on a specific domain problem to develop another model with the aim of solving a different type of domain problem.

Transfer learning is the use of a trained model on a particular task repurposed for a different but similar task (Brownlee, 2019). The use of transfer learning can present enormous benefit as it can shorten the amount of time it takes to train and develop new a model, also does not necessarily require large amount of data as the model is already pre-trained, especially when training data is limited. Transfer learning techniques are used by researchers to solve tasks because of the numerous benefits they provide, the most important of which is time savings when developing a model. This is due to the fact that rather than creating a new model from scratch, which could take time depending on the complexity of the domain problem. As a result, a pre-trained model developed or trained on a similar problem can be used. However, for transfer learning to be effective it must be applied to related or similar problems. That is, old task and new tasks should be similar. Otherwise, it can result in poor performance of the model which is usually referred to as negative transfer. Currently one of the challenges being faced with application of transfer learning technique is that there is no defined, specific standard or algorithms which determines how tasks are similar or related.

Some major pre trained deep learning models for computer vision are

- Inception v3
- VGG- 16
- VGG- 19
- ResNet-50
- Xception

We briefly discuss some of the popular pre trained architecture highlighting some its important features.

3.2.1 VGG-16 ARCHITECTURE

VGG-16 (Visual geometry group 16) is a CNN model used mainly for computer vision tasks such as object detection, image segmentation and image classification. This model was developed by University of Oxford. The vgg-16 architecture which is illustrated in Figure 3.2 is characterized by small 3x3 filters and 16 layers which includes 13 convolutional layers and 3 fully connected layers and 1 softmax layer. This architecture can be broken down into 5 blocks of layers with a different number of convolutional layers.

- The first block consists of 2 convolutional layers, each followed by a max-pooling layer.
- The second block also consists of 2 convolutional layers, each followed by max-pooling layers
- The third block consists of 3 convolutional layers, each followed by max-pooling layer.
- The fourth block consist of 3 convolutional layers, each followed by a max-pooling layer.
- The final block consists of 3 fully connected layers, with the final layer producing the output of the network model.

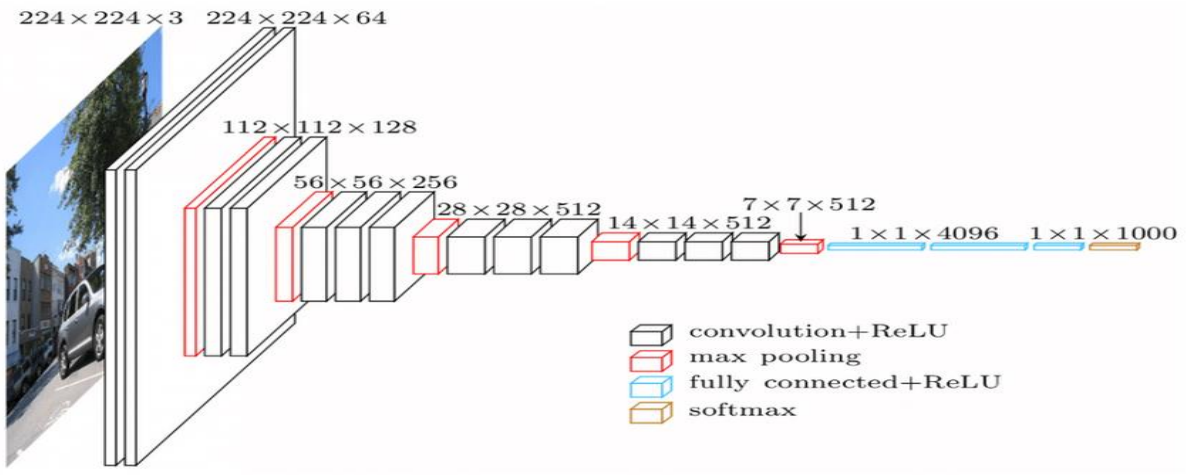


Figure 3.2: showing of basic VGG-16 architecture (Loukadakis et al., 2018)

3.2.2 VGG-19 ARCHITECTURE

VGG-19 is a convolutional neural network (CNN) model which is a variant of visual geometry group (VGG), this model which is illustrated in Figure 3.3 was developed at the University of Oxford 2014, its architecture is characterized by use of small (3x3) filters and 19 layers (16 convolution layers, 3 fully connected layers, 5 MaxPool layers and 1 Softmax layer). The architecture is divided into two parts.

- The first part is made up of multiple convolutional layers, max pooling layers with RELU activation function. The convolutional layers use filter size of 3x3 and a stride and padding of 1, which helps to preserve the spatial dimensions of the given input. The max pooling layers reduce the spatial dimensions of the feature maps as this helps to reduce the computational cost of the network.
- The second part is made up of fully connected layers which accepts outputs of the convolutional layers and perform classification tasks. The fully connected layers use a large number of neurons which helps to capture the high-level features of the input.

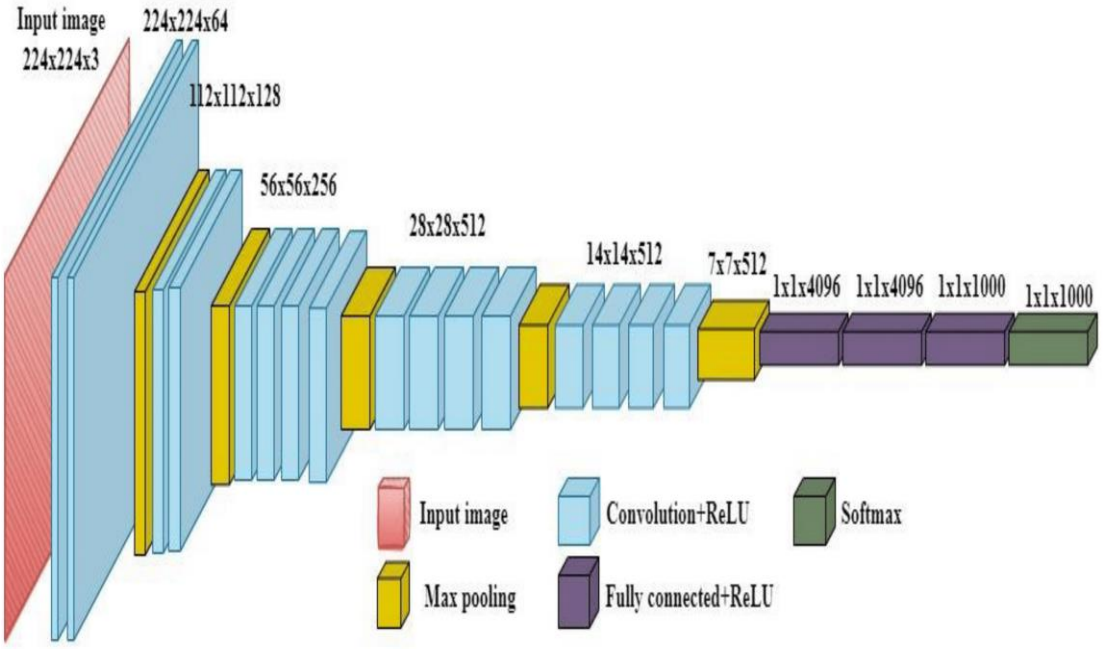


Figure 3.3: showing VGG-19 architecture contain different layers (Nguyen et al., 2022)

3.2.3 RESNET-50 ARCHITECTURE

This model is a type of CNN architecture which belongs to the ResNet (Residual Network) family of model. This CNN architecture is one of the most popular and widely used, which was introduced by Microsoft research in 2015. This architecture illustrated in Figure 3.4 addresses the challenges often associated with deep neural networks which is the problem of

vanishing gradients during training, where the gradients become incredibly small as they propagate backwards through many layers making training difficult. Hence, the architecture utilizes residual connections also referred to as skip connections or shortcut connections. The residual connections allow for the architecture to learn the residual function which simply is the difference the input and the expected output. Instead of attempting to directly learn the complete mapping, the network learns to approximate the residual outcomes. These connections make it possible for the network to effectively pass information from lower layers to higher layers which enable the model to learn more efficiently and effectively perform better. The building block of the ResNet-50 architecture is the residual block which consists of two or more convolutional layers with the addition of skip connections which can bypass these layers. The skip connection has the ability to directly propagate the inputs from one layer to the next layer, thereby allowing the gradient to flow easily during backpropagation. With these shortcuts or skip connections introduced, ResNet-50 is effectively able to train very deep networks with its 50 layers present.

The ResNet-50 is applied to a wide range of computer vision tasks such as object detection, image classification and semantic segmentation. The architecture is characterized by its residual connections which are designed to make it easier for the network to identify functions.

The ResNet -50 architecture consists of 50 layers which includes:

- The input layer: this layer accepts the input image size of 224 x 224 pixels
- The convolutional layers: these layers are responsible for extracting features from the input image. ResNet-50 has several convolutional layers with different filter sizes and strides.
- The pooling layers: these layers are used to reduce the spatial dimensions of the feature maps, which helps to reduce the computational complexity of the network
- The residual layers: these layers are the main feature of the ResNet architecture, these connections are designed to make it easier for the network to learn identity functions.
- The fully connected layer: this layer is used to make the final prediction. It is connected to all the neurons in the final feature map, so it can make use of all the features that were extracted by the convolutional and pooling layers
- The output layer: this layer produces the final prediction of the network.

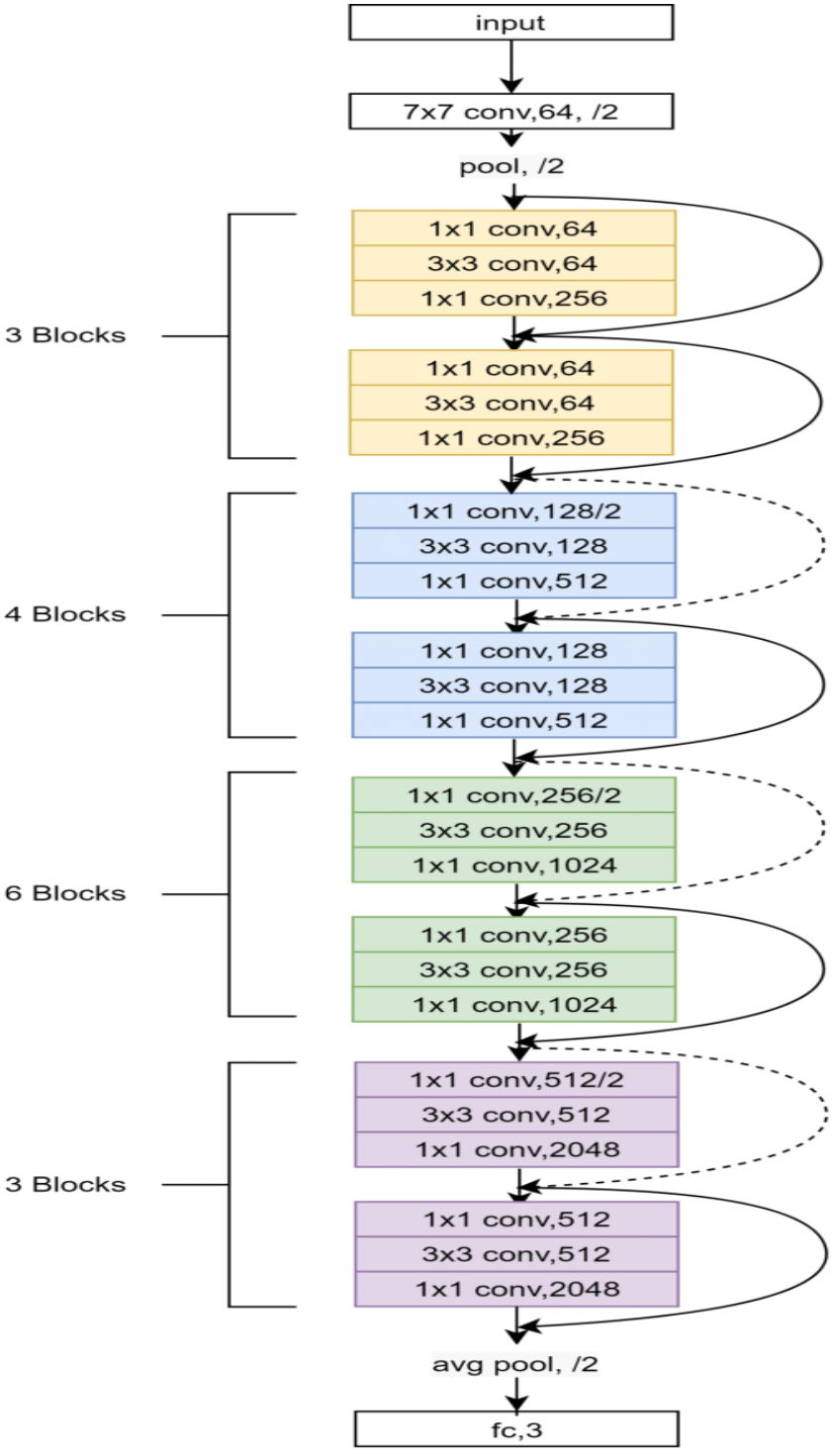


Figure 3.4: Showing the an illustration of the ResNet -50 architecture with skip connections (Al-Humaidan & Prince, 2021)

3.3 MALWARE VISUALIZATION

Due to the increasing rate of malwares attack and the increasing level of sophistication, this is as a result of the fact that most malware authors usually modify small sections of the existing malware codes manually or using automation tools to produce newer malwares. It has become crucial to find effective methods for understanding and combating these threats. One approach gaining traction is malware visualization, which leverages visual similarities properties. This is evident when malware samples are visualized which reveals similarities in structure, composition and other crucial feature information. Hence visualizing malwares can be used to quickly classify malwares into groups. One of the first researcher to use visualization was (Nataraj et al., 2011) who represented malware samples as grayscale images in order to distinguish similarities and differences among malwares samples. This showed visual similarities of malwares belonging to the same family.

Malware visualization is an effective means of representing malware samples in a visual form as this provides the unique opportunity to employ image processing techniques for the detection and classification of malwares. It shows the structure and composition of the malwares as well as other feature information. Visualizing malwares as images unlocks several advantages. Firstly, it allows security analysts to perceive patterns and similarities in the visual representation of malware, which may not be immediately apparent when examining raw code. These visual patterns can serve as valuable indicators for grouping and classifying malware samples efficiently. Moreover, visualizations enable analysts to identify modifications made by malware authors to existing code, whether manually or through automation tools. This insight aids in understanding the evolving nature of malware and devising effective defence strategies.

By representing malware samples as images, the opportunity to use image processing techniques in malwares detection and classification arises. Malware images typically have both local and global features or descriptors. Local features or descriptors are tiny patches or pixels within the image. These localized attributes enable us to discern fine-grained details, such as specific code segments or unique pixel patterns. By examining these localized features, we can identify commonalities among malware variants that share similar code sections or visual characteristics. While global feature or descriptor gives a general or holistic description of the whole malware image. They encompass contour, shape, and texture representations that encapsulate the overall structure and appearance of the malware. Analysing these global features allows us to capture

the overarching characteristics of malware, facilitating higher-level comparisons and classifications.

By combining local and global features, malware visualization empowers security researchers to gain a comprehensive understanding of malicious code. It enables the detection of similarities and patterns that may not be immediately apparent through traditional analysis methods alone. Moreover, visualizing malware facilitates the development of more effective and targeted defence mechanisms, as researchers can leverage these insights to devise advanced detection and prevention strategies.

Malware visualization is a powerful approach for unravelling the complexities of malware attacks. By transforming malware samples into visual representations and leveraging image processing techniques, we can identify shared structures, classify malware into distinct groups, and develop robust defence mechanisms to safeguard against evolving threats.

3.4 E-CNN METHODOLOGY

The concept of using visualizing malwares for classification is not a new technique as it has been done by various researchers. However, there is an imperative or need to improve and develop a better or effective model for detection and classification of malwares using newer methods as well as latest dataset of malware samples. This is achieved by developing a deep enhanced CNN model called E-CNN. Although, existing research has laid the groundwork for malware visualization, our approach aims to push the boundaries further by integrating cutting-edge techniques. The E-CNN model will capitalize on the inherent advantages of CNNs, such as their ability to automatically learn hierarchical features from raw input data. By leveraging this deep learning architecture, our model will gain a deeper understanding of the complex visual patterns within malware samples, thereby enhancing its classification accuracy.

The followings stages are outlined in developing a model namely: Dataset Preparation,

Visualized malware pre-processing, feature Pre-processing and classification (Hammad et al., 2022)

3.4.1 MALWARE IMAGES

In order to capture the inherent characteristics of malware binary files, the binaries are visualized as RGB images so as to extract the texture and coloured features enabling clear distinction between different malware binaries. By representing malwares as images, we

can effectively identify clear feature distinctions (both local and global descriptors) from malware binaries of special byte sequences which consist of Dynamic Link Libraries (DLLs), string constraints, uninitialized data, debug information which are present in the code section and data section as well as other sections. The visual representation of malware binaries offers valuable insights into their structure and composition. By converting the complex binary code into images, we gain a more intuitive understanding of the visual and structural characteristics shared by malwares of the same type or belonging to the same family. This visual similarity becomes evident when observing the resulting images, this is shown in the Figure 3.5 and 3.6.

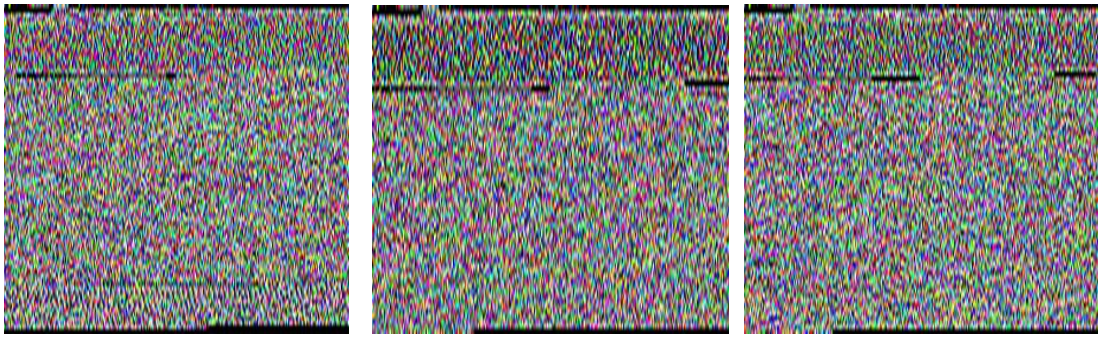


Fig 3.5 Images of malware samples belonging to Allaple Family

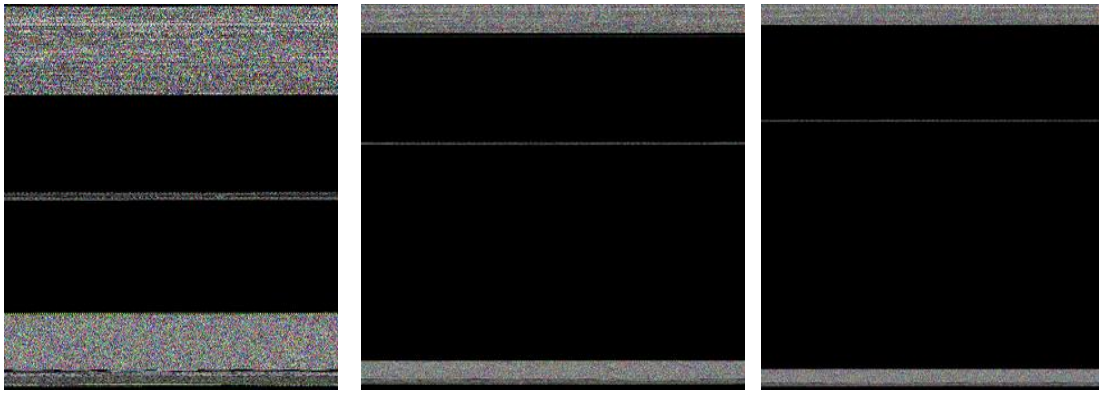


Fig 3.6 Images of malware samples belonging to Autorun Family

When analysing malware images, we can observe common patterns and visual cues that indicate shared traits among related malware samples. These similarities can manifest in the form of recurring pixel arrangements, distinct colour distributions, or recurring shapes and contours. Hence, this relationship can further investigate and quantify these shared characteristics, facilitating the classification and grouping of malwares into distinct families or types.

3.4.2 DATASETS

3.4.2.1 MALEVIS DATASET

The dataset used in the development of the E-CNN model is called the Malevis dataset which is an open set image dataset which consist of 26 classes of byte images (25 malware classes and 1 legitimate class) as shown in the Table 3.1. This dataset was constructed by extracting binary images from malware files in 3 channels RGB form by bin2png script developed by Sultanik. The generated images are then resized into two different squared size resolution (224x224 and 300x300 pixels). The Malevis dataset consist of 9100 training and 5126 validation RGB images.

Table 3.1: Description of the Malevis Dataset.

S\N	Family Name	Family	Total Samples
1	Adposhel	Adware	494
2	Agent	Backdoor	470
3	Allaple	Worm	478
4	Amonetize	Adware	497
5	Androm	Backdoor	490
6	BrowseFox	Adware	493
7	Dinwod	Trojan	499
8	Elex	Adware	500
9	Expiro	Virus	501
10	Fasong	Trojan	500
11	HackKMS	Hacktools	499
12	Hlux	Worm	500
13	Injector	Trojan	495
14	InstallCore	Adware	500
15	MultiPlug	Adware	499
16	Neoreklami	Adware	500
17	Neshta	Virus	497
18	Other (legitimate)	Legitimate	1832
19	Regrun	Trojan	485
20	Sality	Virus	499
21	Snarasite	Trojan	500
22	Stantinko	Trojan	500
23	VBA	Virus	500
24	VBKrypt	Trojan	496
25	Vilsel	Trojan	496
26	Autorun	Worm	496

3.4.2.2 EXPERIMENTAL BENCHMARK DATASET

In other to evaluate the newly developed E-CNN model performance, a public and popular benchmark malware dataset called Malimg is used. The dataset provides a diverse collection of malware images from different families, making it suitable for evaluating the robustness and generalization ability of classification models. Many researchers have used the Malimg dataset as a benchmark to compare the performance of their models with existing state-of-the-art methods. This dataset contains 9435 grayscale image samples from 25 malware families as shown in the Table 3.2. This will further test the suitability of the E-CNN model.

Table 3.2: Data Description of Malimg Dataset

S/N	Family Name	Family	Total Samples
1	Allaple.A	Worm	2949
2	Allaple.L	Worm	1591
3	Adialer.C	Dialer	122
4	Agent.FYI	Backdoor	116
5	Alueron.gen!J	Trojan Horse	198
6	Autorun.K	Worm AutoIT	106
7	C2LOP.gen!g	Trojan Horse	146
8	C2LOP.P	Trojan Horse	200
9	Diaplatform.B	Dialer	177
10	Dontovo.A	Trojan downloader	162
11	Fakerean	Rogue	381
12	Instantaccess	Dialer	431
13	Lolyda.AA1	Password Stealer	213
14	Lolyda.AA2	Password Stealer	184
15	Lolyda.AA3	Password Stealer	123
16	Lolyda.AT	Password Stealer	123
17	Malex.gen!J	Trojan Horse	136
18	Obfuscator.AD	Trojan downloader	142
19	Rbot!gen	Backdoor	158
20	Skintrim.N	Trojan	80
21	Swizzor.gen!E	Trojan downloader	128
22	Swizzor.gen!I	Trojan downloader	132
23	VB.AT	Worm	408
24	Wintrim.BX	Trojan downloader	97
25	Yuner.A	Worm	800

3.4.3 MODEL OVERVIEW

This research work proposes a convolutional neural network CNN for classification of malwares, this developed model is referred to as deep enhanced CNN (E-CNN). This model architecture is developed with the aim of achieving high accuracy for classification of malwares into different classes and to ensure that the model is generic data independent and learns the discriminative feature representation from the image data itself. The parameters which were considered using an optimization library called Keras Tuner library which iteratively fine tunes the CNN layers considering different hyperparameters until best performance values are discovered, such hyperparameters includes the activation function which is Rectified Linear Unit (ReLU) for different layers, this is a non-linear function which is very effective identifying complex relationships within data when analysed. Also, SOFTMAX for the last layer of the model as well as using categorical cross entropy loss function for multi-classification of output classes or labels. Unlike other optimizers, Adam optimizer was selected. This is because of its superior performance and ability to establish adaptive learning rates for each parameters (El-Shafai et al., 2021). Furthermore, ADAM optimizer which tries to minimize the loss during use of training data and with categorical cross entropy to train the model for classification of malwares into different malware classes as well as legitimate class. The Figure 3.7 shows the model diagram which consist of the different layers and Figure 3.8 shows the schematic architecture of the E-CNN model is shown below.

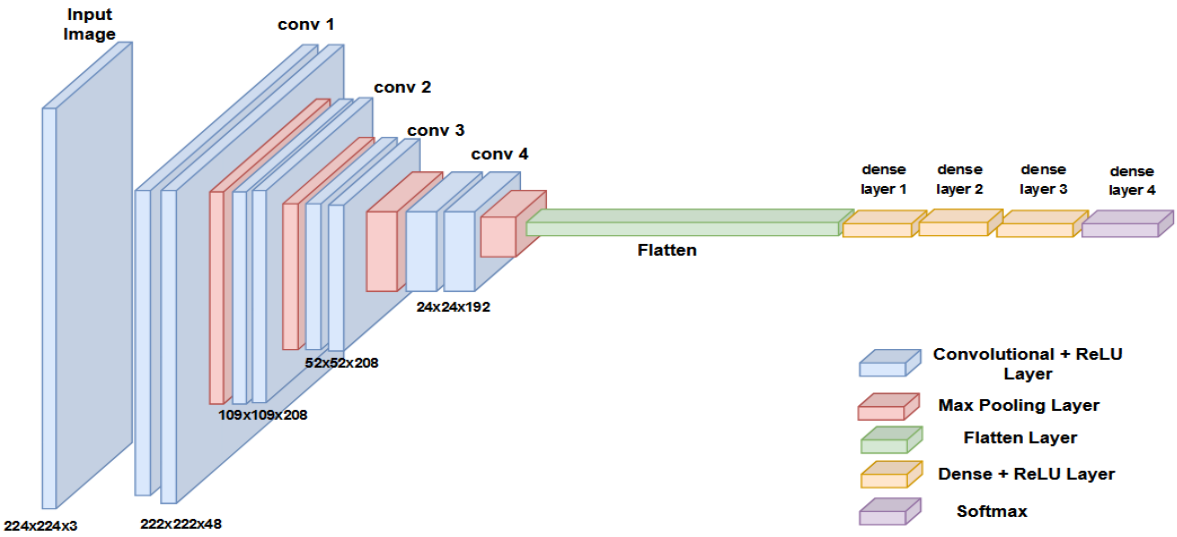


Figure 3.7: Overview of the E-CNN model architecture

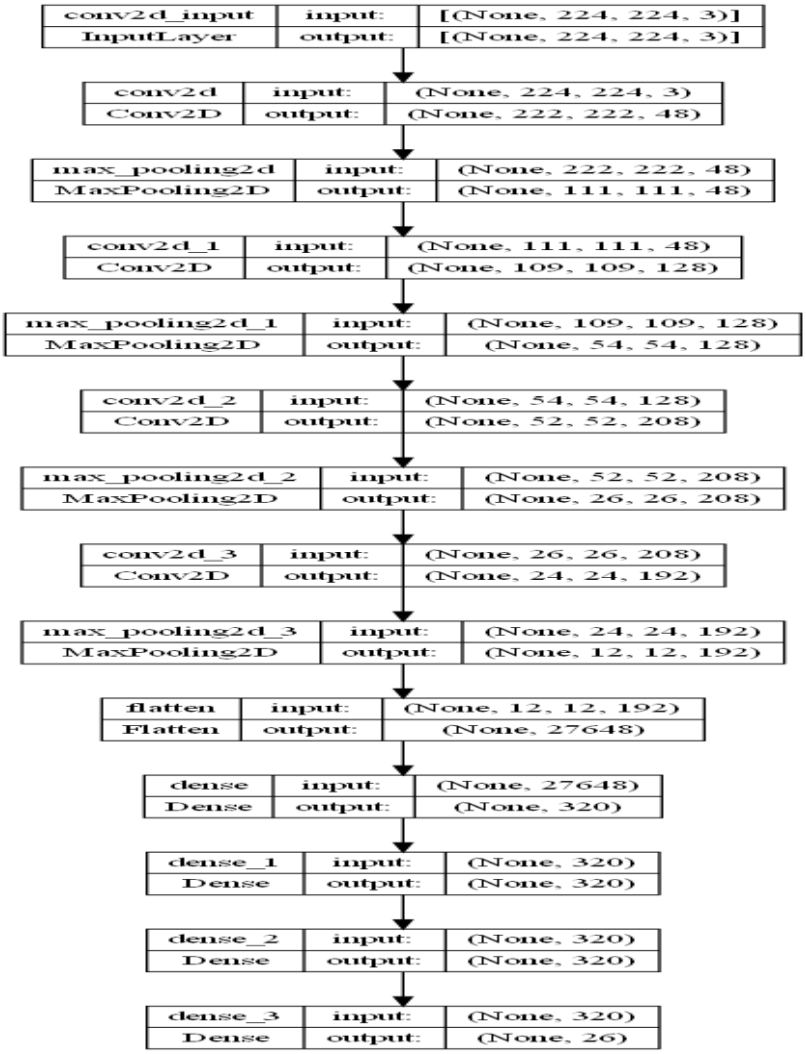


Figure 3.8: Schematic Diagram of the E-CNN architecture

3.5 PERFORMANCE EVALUATION METRICS

Performance metrics is important in evaluating the performance of E_CNN model as this assess how effective the model is in classification of malwares. The performance evaluation metrics includes

1. Accuracy Score
2. Precision score
3. F1- score
4. Recall
5. Confusion Matrix

3.5.1 ACCURACY SCORE

This refers to the ratio of correctly predicted outcomes to the total number of possible outcomes. The accuracy score metrics calculates the percentage of correct predictions made by the classifier. As a result, the overall correctness of the classifier is determined. The accuracy formular is stated equation 3.1.

$$\text{Accuracy} = \frac{True\ Positive + True\ Negative}{Total\ Outcomes} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (3.1)$$

3.5.2 PRECISION SCORE

This is the ratio of correctly predicted positive outcomes to the total predicted positive outcomes. The precision score represents or measures the percentage of correct positive predictions as shown in equation 3.2.

$$\text{Precision score} = \frac{True\ Positve}{True\ positive + False\ positive} \qquad \ldots\ldots\ldots\ldots.. (3.2)$$

3.5.3 RECALL

This refers to the ratio of correctly predicted outcomes to the overall outcomes in the positive class. Recall is the proportion of real positive cases that are correctly predicted positive. This represents the number of true positives divided by the total number of true positives and false negatives as stated in equation 3.3

$$\text{Recall} = \frac{True\ Positve}{True\ positive + False\ negative} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.3)$$

3.5.4 F1 SCORE

F1 score also referred to as F measure, it is the weighted average of the recall and precision. it is the harmonic mean of precision and recall. It represents the harmonic mean of the precision and recall scores, which is used to calculate the F1 score. An F1 score of 1 is assigned to a model that has perfect precision and recall scores. This formular is stated in equation 3.4

$$\text{F1} = 2\ X\ \frac{Precision\ X\ Recall}{Precision + Recall} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (3.4)$$

3.6 STEPS/ PROCEDURE OF PROPOSED RESEARCH MODEL

1. The dataset consists of both legitimate and malware samples represented as images with a total of twenty-six classes with twenty-five (25) malware classes and a legitimate class). It is with the Malevis dataset the proposed E-CNN malware classification model is developed. The image samples give a visual representation of the feature and texture information of the samples. This allows image samples to be classified according to the family they belong as they are often similar visually.
2. The image samples have a dimension of 224x224 pixel as input for the model, then the E-CNN model is then defined using optimization functions which determines the best hyperparameters values such as number of layers (convolutional, pooling, dense layers,), kernel size and filters. So as to determine the best model for classification of malwares.
3. The E-CNN model having being developed is further tested with a test dataset which consist of images samples of different malware families and legitimate files. The model develop in step 2 are evaluated using the performance metrics which includes accuracy score, precision score, recall, and f1 score.
4. The transfer learning technique is applied on the Malevis dataset on the existing pre-trained models which includes (VGG19, RESNET50, VGG16). This process is done until optimal performance values is determined.
5. The pre-trained model's evaluation metrics results are compared with E-CNN model so as to access the performance of newly developed model (E-CNN) for malware classification.
6. The E-CNN model and the pre-trained models are test with public benchmark dataset (MALIMG),

3.6.1 ALGORITHM

Algorithm: To Develop E-CNN model

Input: Using Malevis dataset to develop, malware classification model

Output: Develop the E-CNN model, using optimized parameters values.

1. Malevis dataset is loaded, which contains visual representation of malware samples and used to get the feature vectors
2. The image samples of the dataset were pre-processed, and the resultant size of the image samples was set at 224 x 224, which serves as input to the model.

3. Using optimization function and techniques to determine the best hyperparameter values in developing the model for malware classification
4. The E-CNN model is developed
5. Using pre-trained model (RESNET50, VGG16, VGG19) on Malevis dataset and access its performance with the newly developed E-CNN model
6. E-CNN model is tested with benchmark dataset (MALIMG).

CHAPTER FOUR

4.0 EXPERIMENTS AND RESULTS

This chapter describes and explains the practical environment in which E-CNN model was developed as well as using transfer learning on the pre-trained models (VGG19, VGG16, RESNET50) on the Malevis dataset and comparative analysis of evaluation metrics of malware classification. The newly developed model (E-CNN) was applied on a public benchmark dataset (MALIMG) of the experiments so as to assess its performance.

4.1 HARDWARE SPECIFICATIONS/PROGRAMMING LANGUAGE

The experiment will be carried out using python language version 3.6,

Some of Python library modules used include tensorflow.keras.image, tensorflow.keras.models keras_tuner, sklearn.metrics, tensorflow.keras.layers, tensorflow.keras.callbacks, matplotlib, pandas, seaborn, numpy.

- Tensorflow library: is an open-source library developed by google for both machine learning and deep learning applications. It offers an end-to-end platform which makes handles numerical computation for development of models.
- Keras library: Keras is a Python-based deep learning API that runs on top of the TensorFlow machine learning platform.
- Anaconda navigation studio: this is an open source software which consists of a collection of packages which is used for data visualization and development both machine learning and deep learning tasks (Anaconda, 2023).
- Matplotlib: this is a comprehensive library for creating data visualization in python (Matplotlib, 2023)
- Pandas: it is a powerful and flexible open source tool used for both analysis and processing of numerical data in python programming language (Pandas, 2023)
- Keras_tuner is an hyperparameter optimization framework which uses search algorithms to find best hyperparameter values.
- Numpy: it is a powerful open-source tool used for n-dimensional arrays vectorization, numerical computation
- Scikit-learn: this is a simple and effective tool used for predictive data analysis for machine and deep learning applications.
- Anaconda Navigator studio

Hardware specification

Dell XPS 8930 Core i7 8th generation,

16GB RAM and 256GB SSD, 32GB GPU

Windows 10 professional 20H2

This experiment is divided into two phases

1. Model development using Malevis dataset and transfer learning
2. Evaluation of model using benchmark dataset Malimg

4.2 MODEL DEVELOPMENT

E-CNN model was developed using Malevis dataset, this involves continuous test and optimizing the values of different parameters which makes up the model. These parameters include convolutional layers, kernel size, optimizers, dense layers, epochs, batch sizes. The E-CNN model consist of 4,252,474 parameters. Also, the parameters of VGG19, VGG16 and ResNet 50 is show in the table 4.1.

Table 4.1: Description showing the model's trainable and non-trainable parameters

	Model Name	Trainable parameters	Non trainable parameters	Parameters
1	**E_CNN(Proposed model)**	**4,252,474**	**0**	**4,252,474**
2	VGG16	138,266	1,735,488	1,873,497
3	VGG19	203,802	10,585,152	10,788,954
4	ResNet 50	53,274	23,587,712	23,638,937

MALEVIS DATASET EXPERIMENT

4.2.1 E-CNN PARAMETERS

The Malevis dataset was used in developing E-CNN model for multi classification of malwares without the use of any data augementation technique, this dataset was divided into two folds or parts, with training set is 80% and testing 20%. The use of validation data is necessary so as to help optimizer and fine tune the model hyperparameters. The following are some of the model

hyperparameter values: epochs=23, batch size = 50, adam optimizer as well as learning rate = 0.0001.

The performance analysis metrics of the model used to evaluate the model were confusion matrix, classification report and well as the learning curves (accuracy and loss) of the model. The model achieved and average accuracy of 83.87 % in classification of the 25 different classes of malwares. The Figure 4.1 shows the accuracy graph of the training and validation data curve.

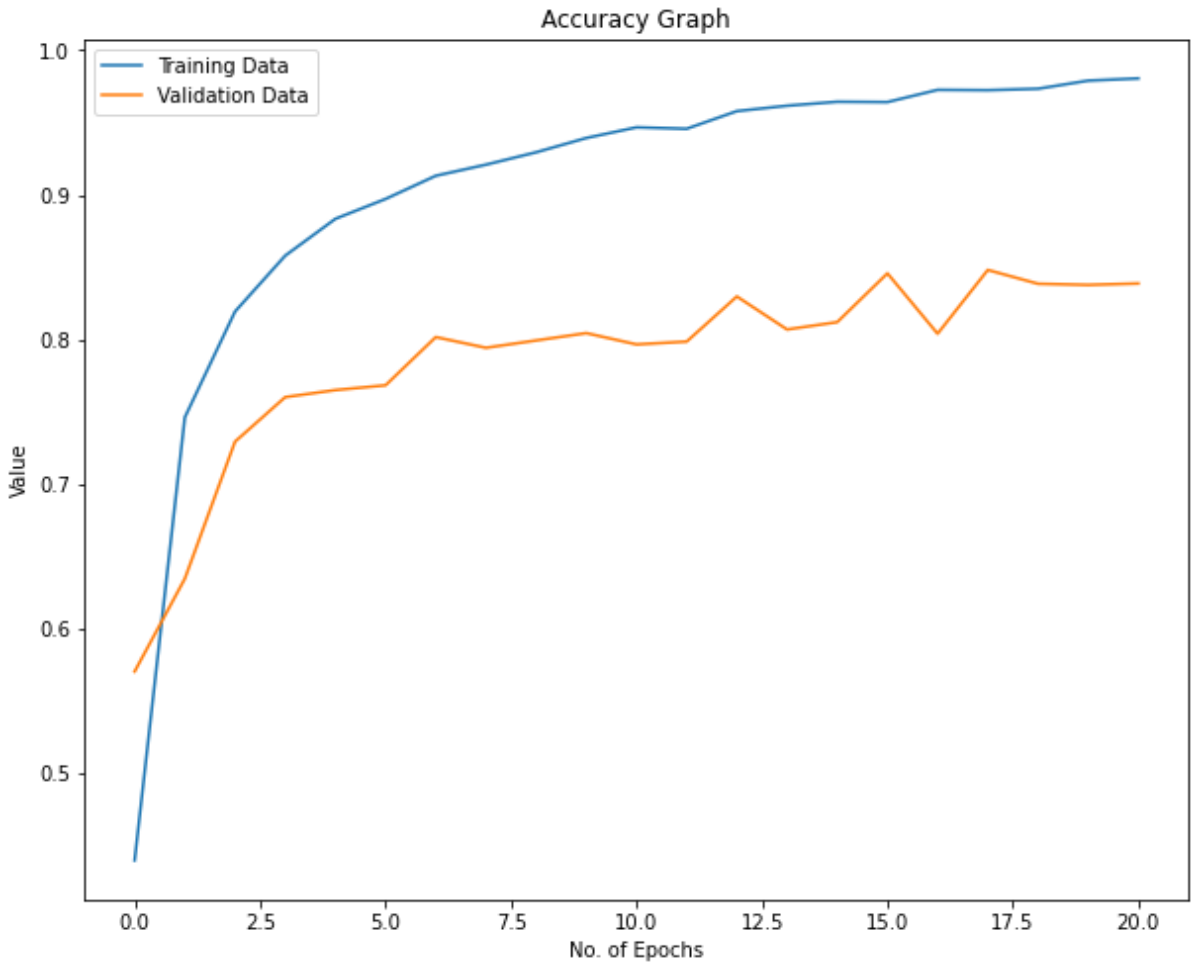


Figure 4.1 Accuracy graph shows Training vs Validation Data curve.

Also, the figure 4.2 shows the loss graph curve of the model for training vs validation data.

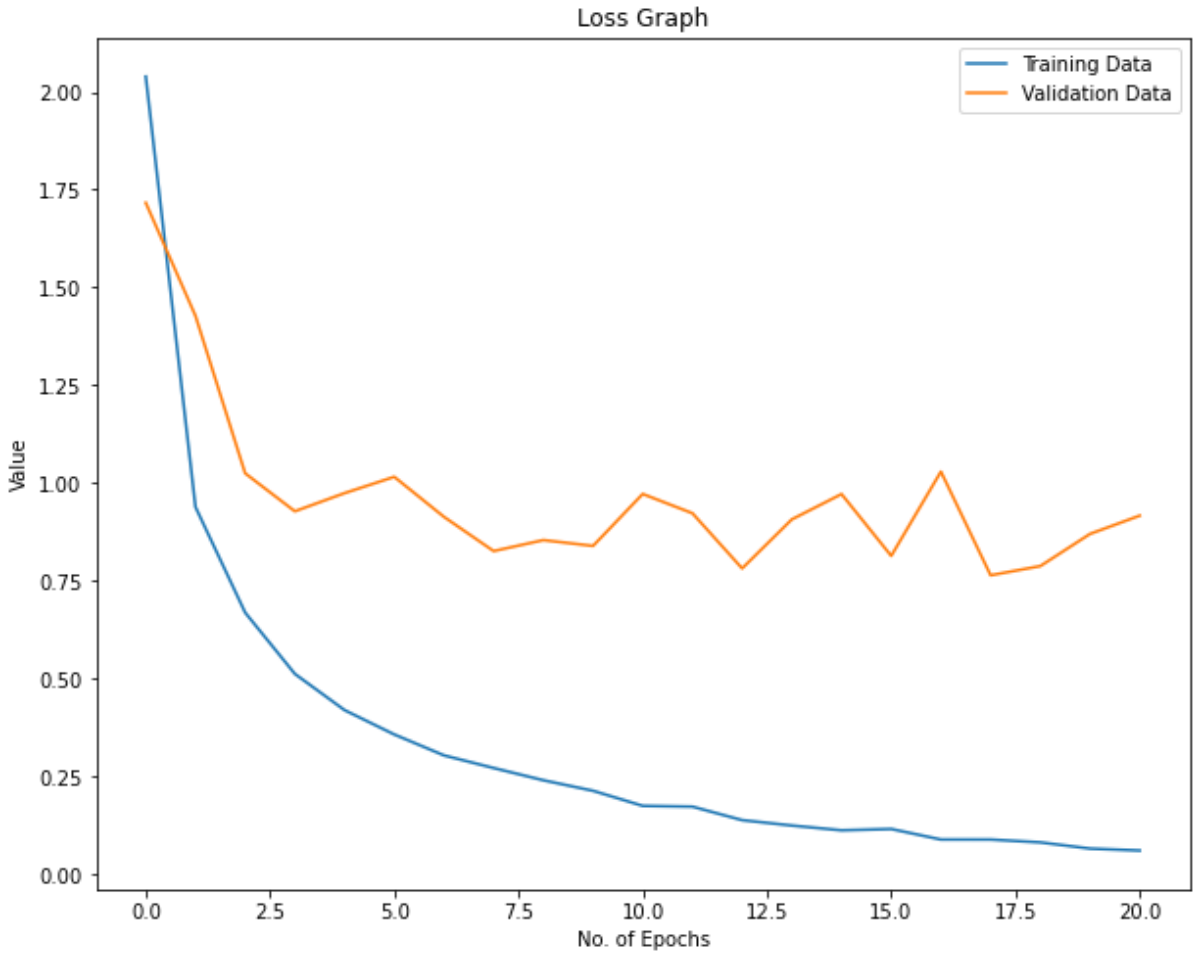


Figure 4.2: Loss Graph showing the training vs validation data curve over a period of 20 epochs

The Table 4.2 below shows the classification report of E-CNN model, accesses the model performance in classifying different malware classes.

Table 4.2: Classification Report of E-CNN on Malevis Dataset

Index	Family Name	Precision	Recall	F1-score	Support
0	Adposhel	0.99	1.00	0.99	144
1	Agent	0.72	0.83	0.77	120
2	Allaple	0.85	0.94	0.89	128
3	Amonetize	0.97	0.97	0.97	147
4	Androm	0.59	0.97	0.74	150
5	Autorun	0.80	0.89	0.84	146

6	BrowseFox	0.89	0.92	0.91	143
7	Dinwod	1.00	0.97	0.98	149
8	Elex	0.79	0.99	0.88	150
9	Expiro	0.79	0.86	0.82	151
10	Fasong	1.00	1.00	1.00	150
11	HackKMS	0.98	0.99	0.98	149
12	Hlux	0.99	1.00	1.00	150
13	Injector	0.72	0.89	0.80	145
14	InstallCore	0.99	0.98	0.98	150
15	MultiPlug	0.92	0.90	0.91	149
16	Neoreklami	0.85	0.99	0.91	150
17	Neshta	0.29	0.62	0.39	147
18	Legitimate	0.96	0.57	0.71	1482
19	Regrun	1.00	0.99	1.00	135
20	Sality	0.42	0.72	0.53	149
21	Snarasite	1.00	1.00	1.00	150
22	Stantinko	0.97	0.97	0.97	150
23	VBA	1.00	1.00	1.00	150
24	VBKrypt	0.60	0.95	0.74	146
25	Vilsel	0.99	1.00	0.99	146
Accuracy				0.83	5126
Macro avg		0.85	0.92	0.87	5126
Weighted avg		0.88	0.83	0.83	5126

The confusion matrix result of the experiment is represented in Figure 4.3 shows E-CNN model performance in predicting different classes of malwares.

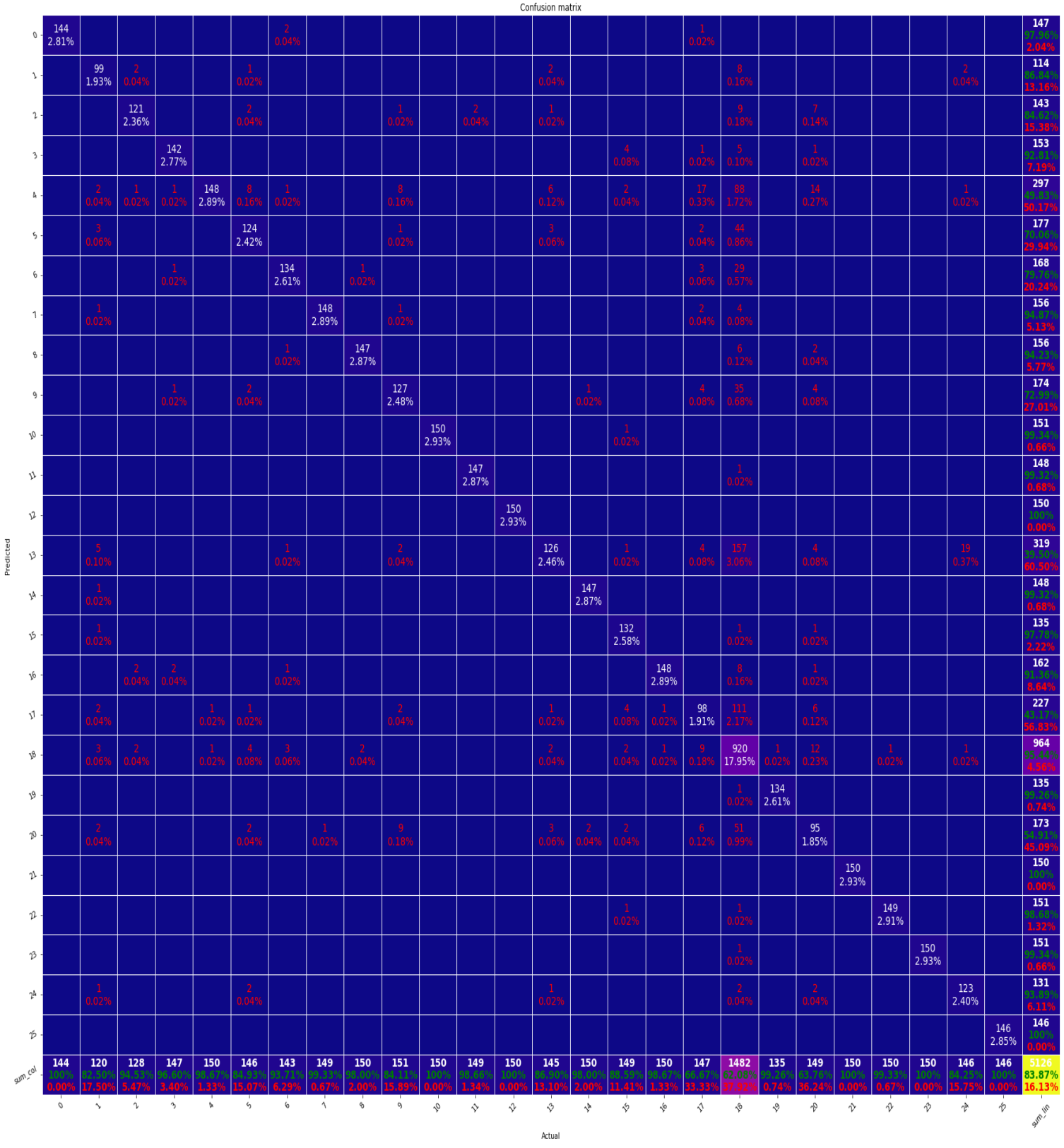


Figure 4.3: Confusion matrix table result of E-CNN model

4.2.2 MALEVIS DATASET TRANSFER LEARNING (VGG16, VGG19, RESNET-50)

Transfer learning was used to train the following CNN architectures: Vgg-16, Vgg19, and Resnet-50. The number of epochs and training time for malware multi classification varies due to differences in architecture size. Applying the Malevis dataset on the resnet-50 architecture yielded an average accuracy score of 49.89% and for loss an average of 2.7 for malware multi classification. The accuracy and loss graph shown in figure 4.4 and figure 4.5 shows the performance of the Resnet-50 architecture when used on training and validation data over 13 epochs (training time).

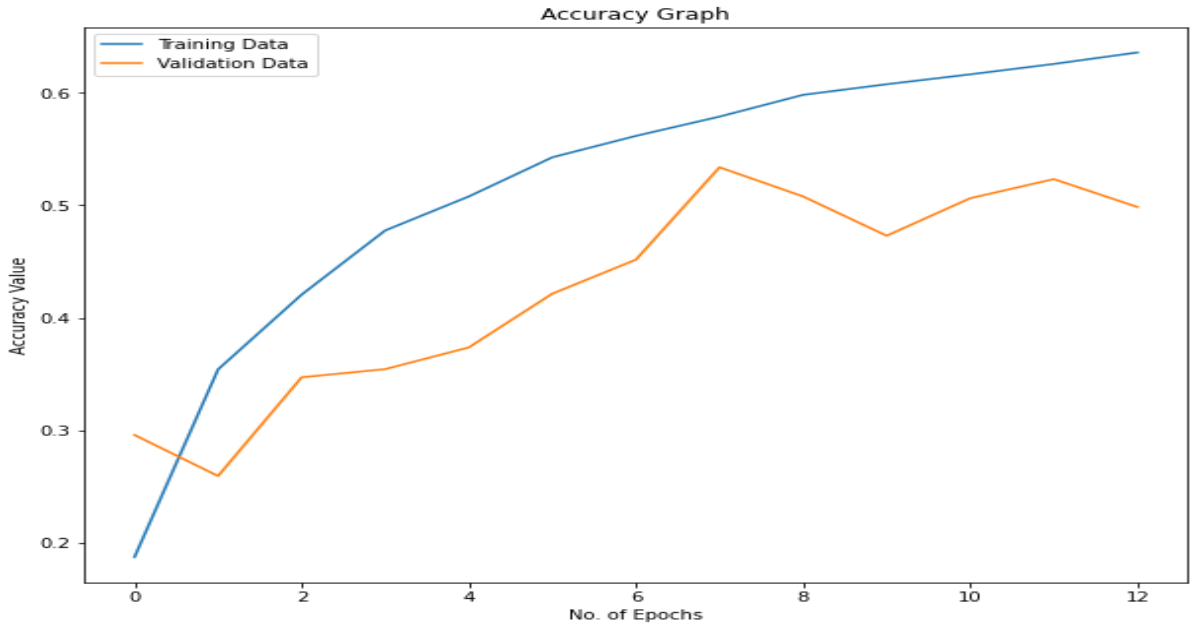


Figure 4.4: Accuracy graph of resnet-50

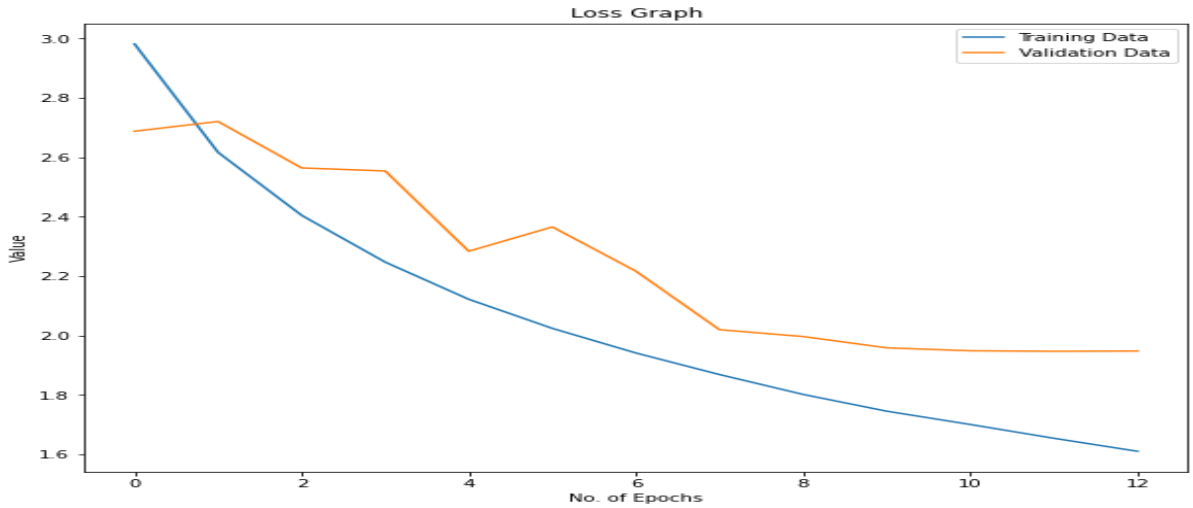


Figure 4.5: Loss graph of resnet-50

The confusion matrix result is shown in Figure 4.6 for Resnet-50 performance

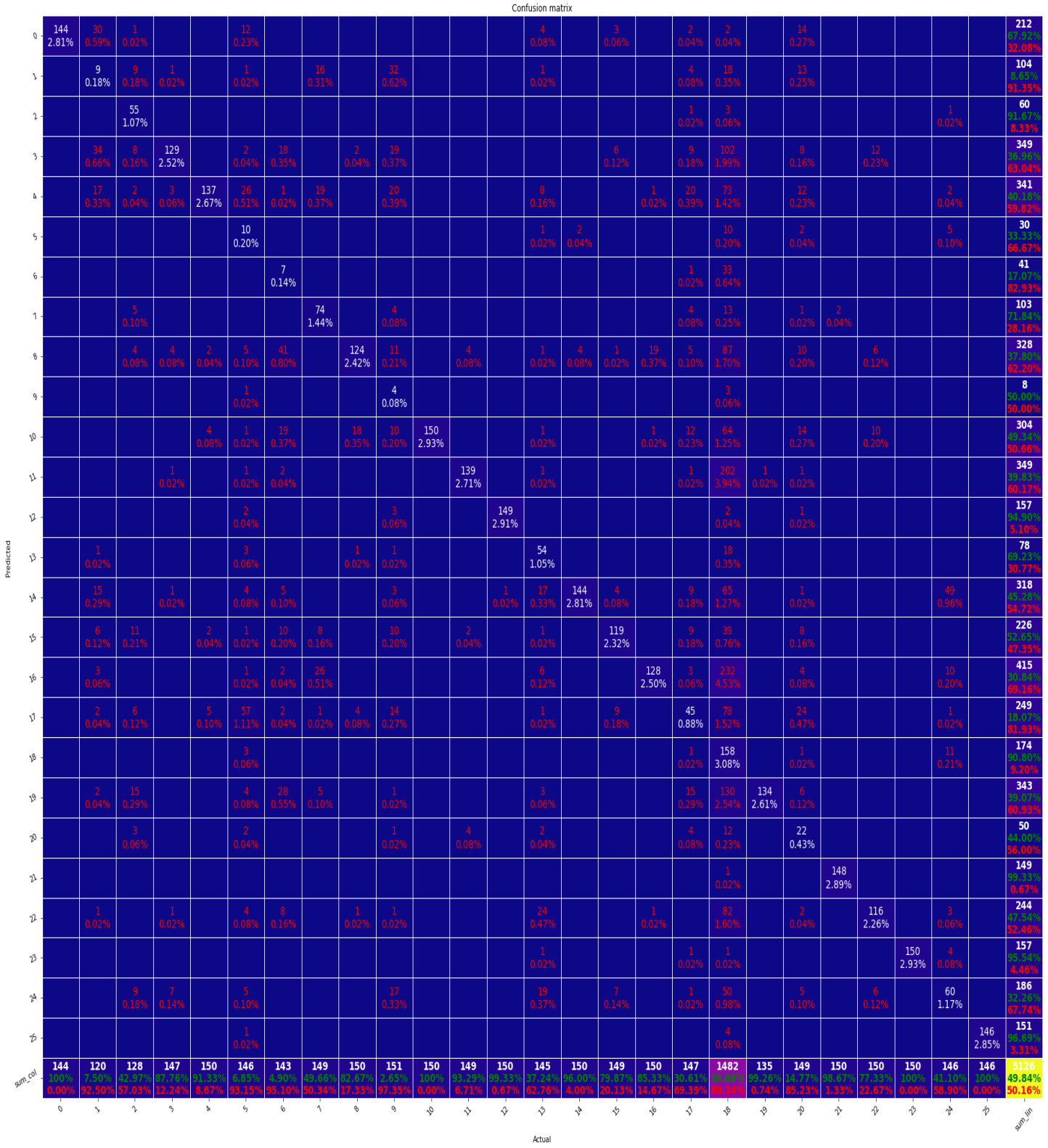


Figure 4.6: showing confusion matrix of malware classification of Resnet-50 architecture

Furthermore, the next CNN architecture VGG16 was trained over 24 epochs. VGG16 architecture achieved an average accuracy score of 82.52% for malware multi classification. The performance of the VGG16 model shows the accuracy and the loss graph in the figure 4.7 and figure 4.8

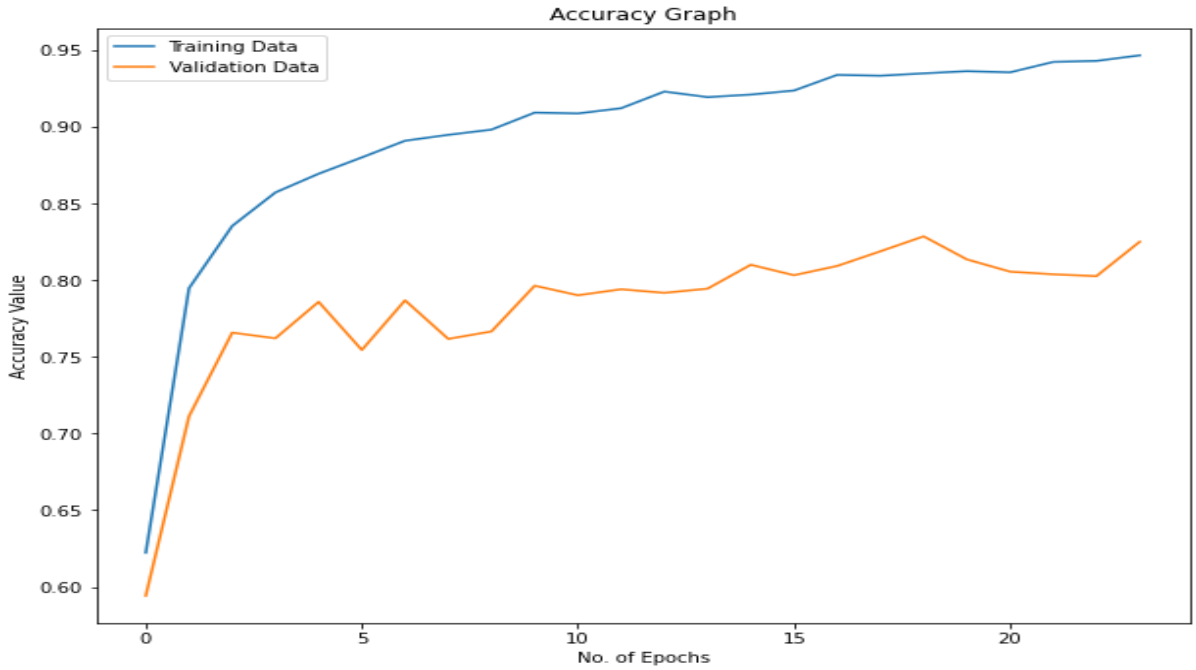


Figure 4.7: Accuracy graph of vgg-16 architecture

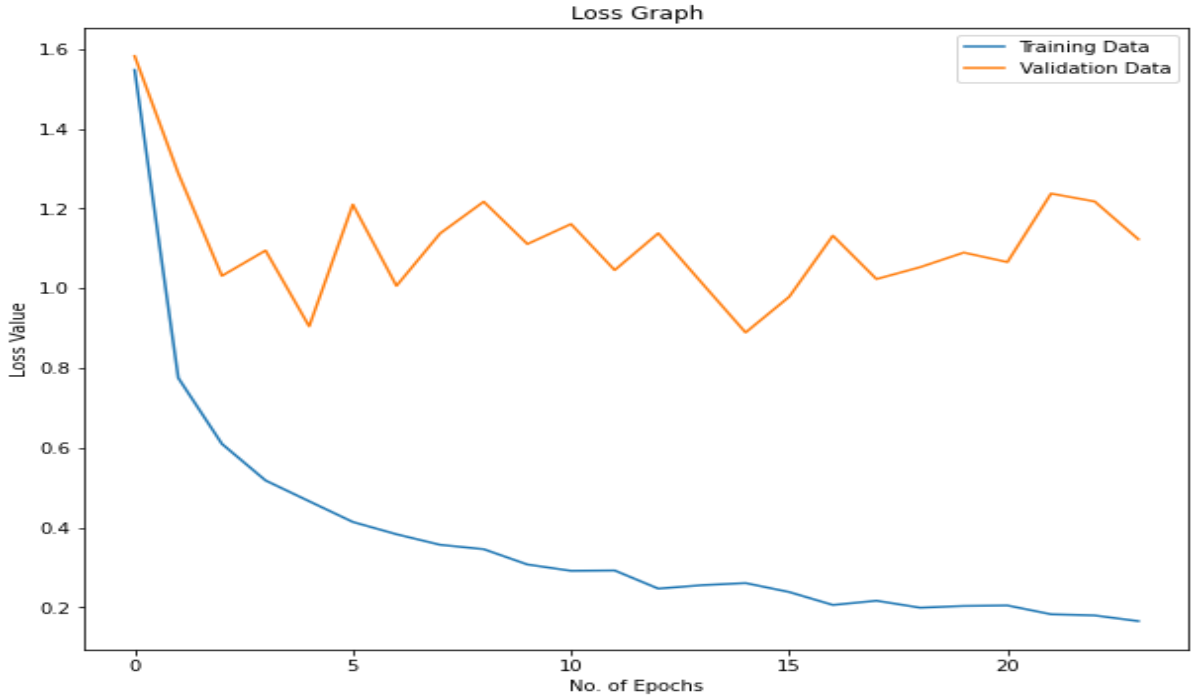


Figure 4.8: Loss graph of vgg-16 architecture

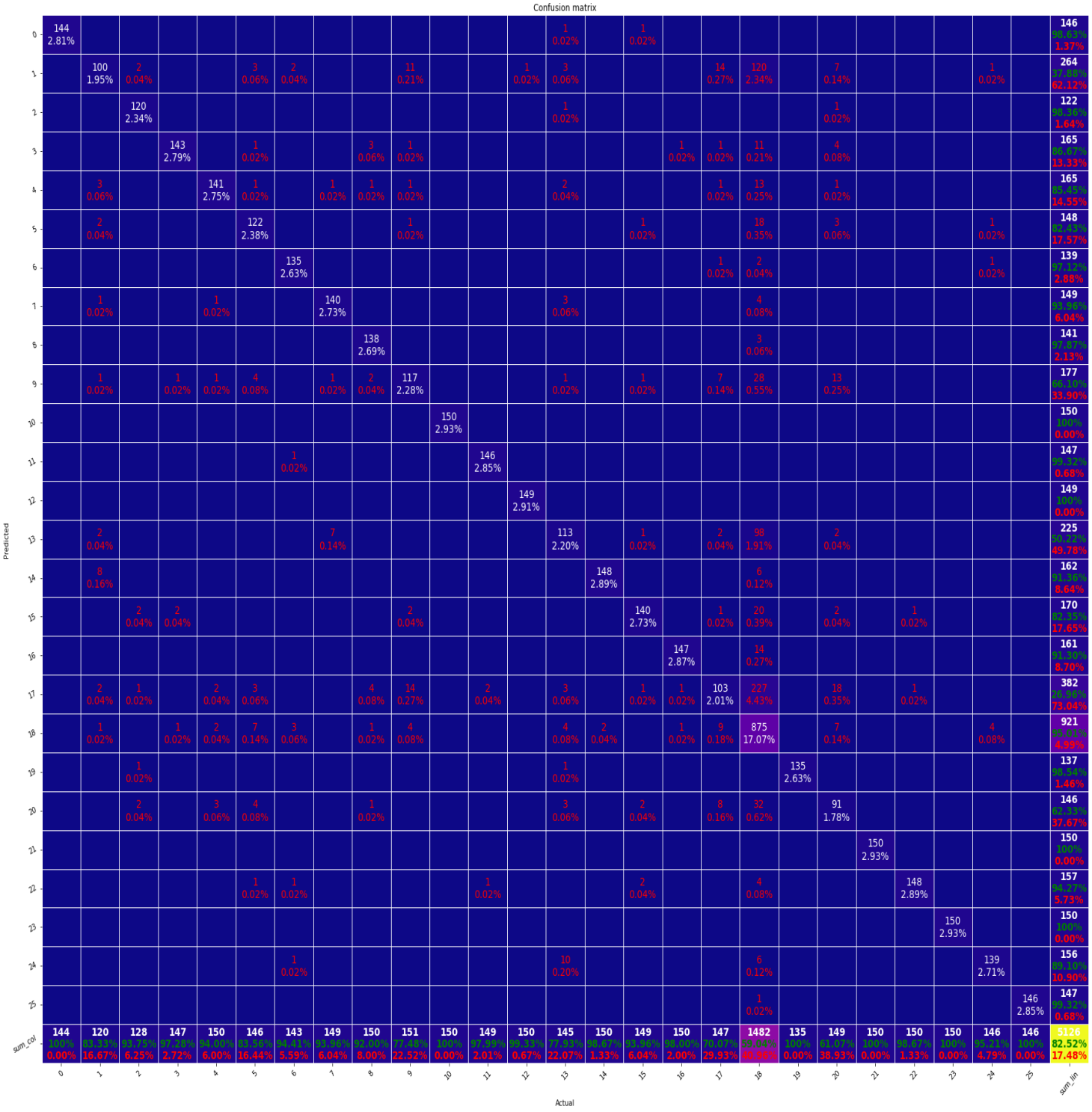


Figure 4.9: showing confusion matrix of malware classification of VGG16 architecture

CNN architecture VGG19 was trained over 23 epochs. VGG19 architecture achieved an average accuracy score of 84.98% for malware multi classification. The performance of the VGG16 model shows the accuracy and the loss graph in the Figure 4.10 and Figure 4.11

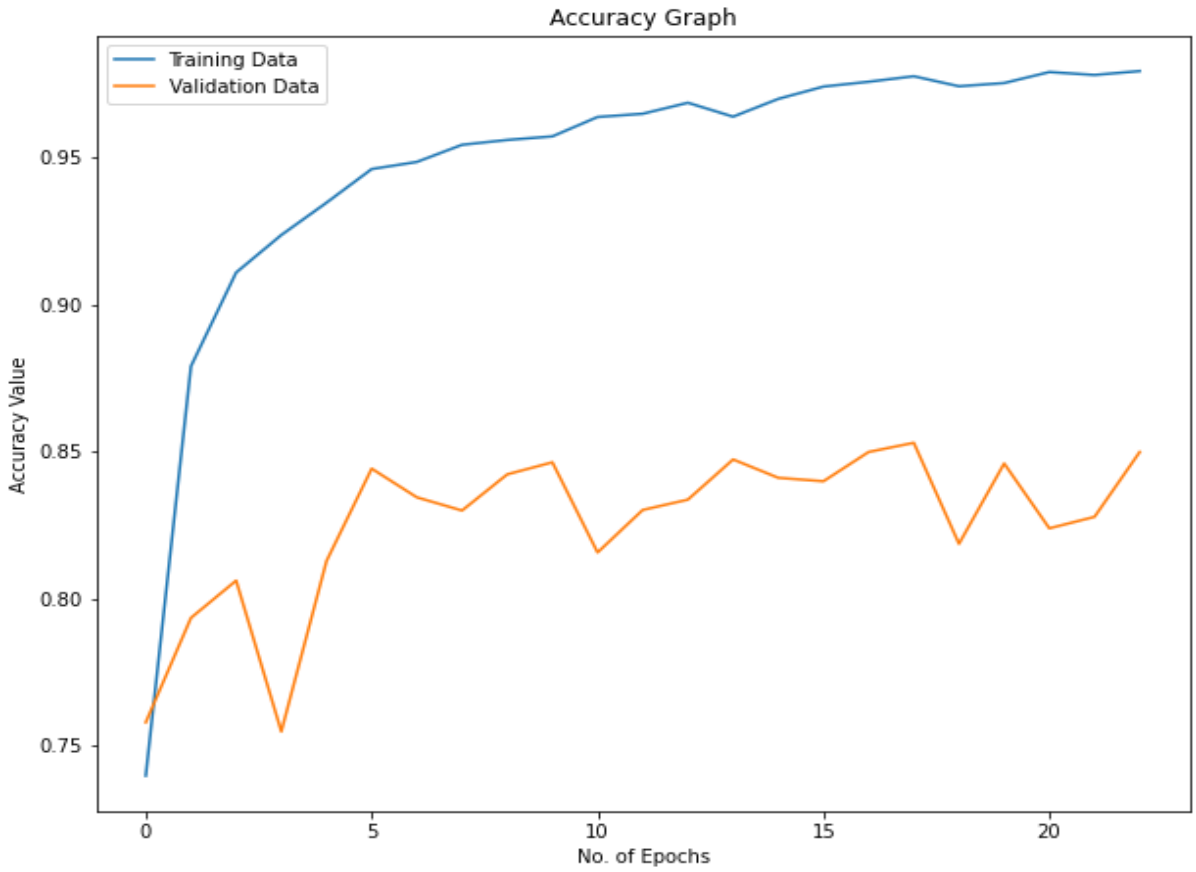


Figure 4.10: Accuracy Graph of malware classification using VGG-19 architecture

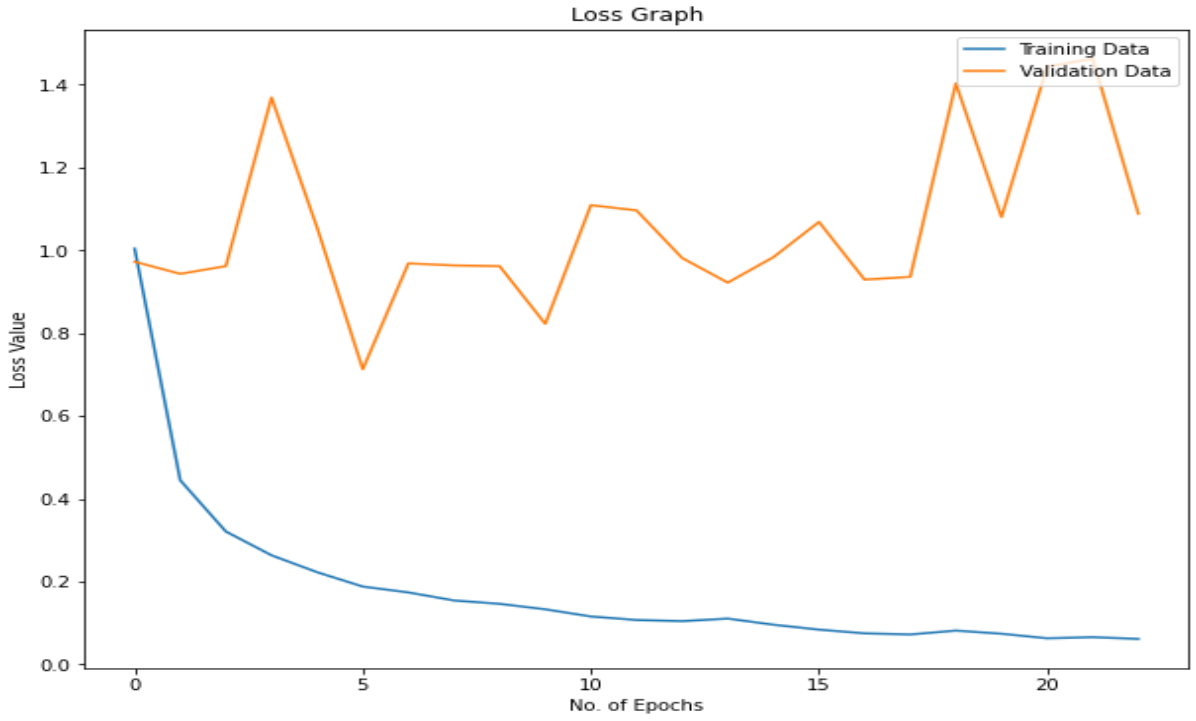


Figure 4.11: Loss Graph for malware classification using VGG-19 architecture

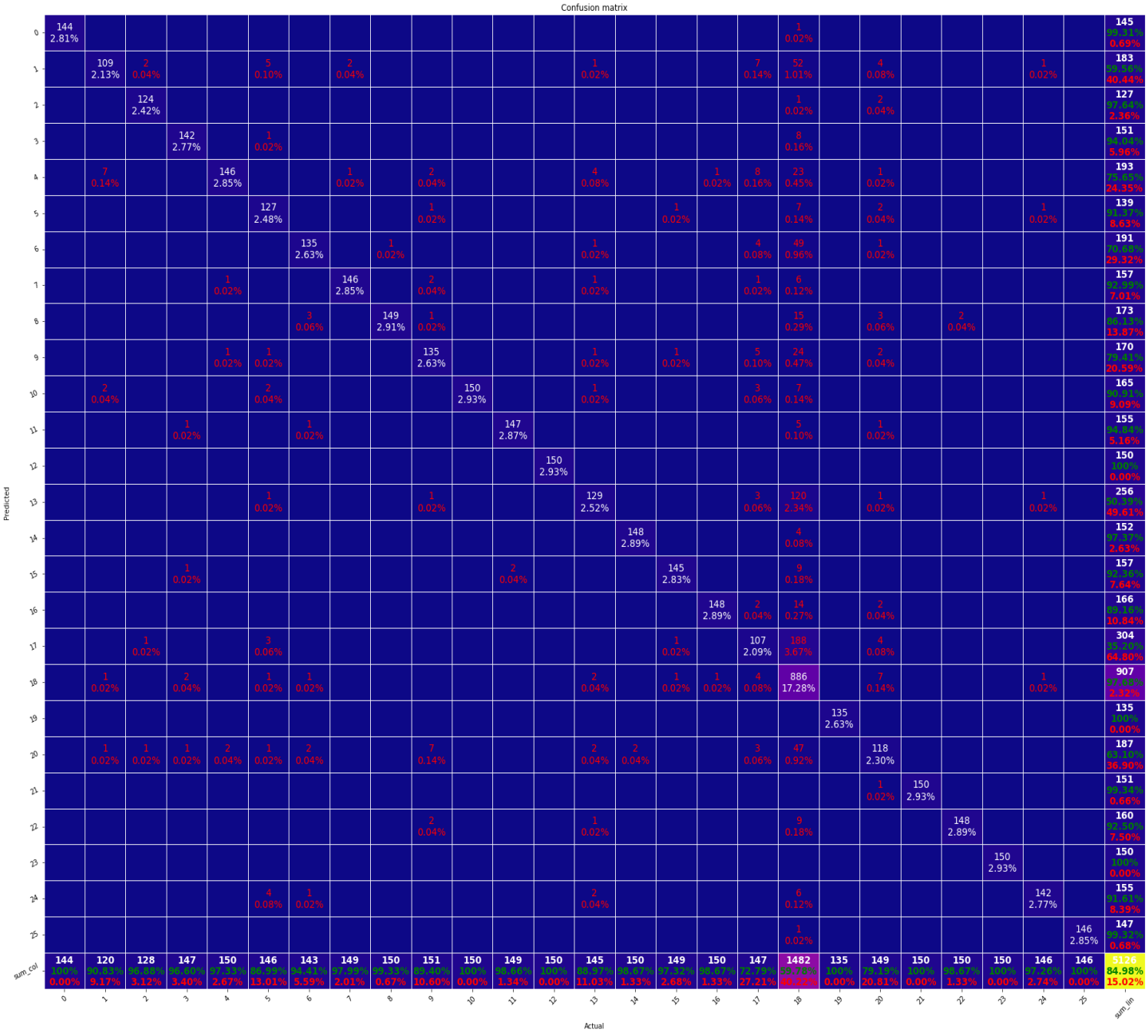


Figure 4.12: showing confusion matrix of malware classification of VGG19 architecture

4.3 RESULT ANAYLSIS

4.3.1 Malevis Dataset Experiment

In the experiment, the E-CNN architecture model for malware classification was developed using optimal hyperparameter values, using the Malevis dataset which is presented in Figure 4.3, fine tuning process was applied to get the best values on both CNN layers and hyperparameters without applying any data augmentation processes with the programming language and hardware

specification explained in section 4.1. The accuracy and loss curves, the confusion matrix and classification report were used as the evaluation metrics. The E-CNN model produced an average accuracy score of 83.87% on the Malevis dataset. Transfer learning techniques were also used to train the following CNN architectures: RESNET-50, VGG-16, and VGG-19, with average accuracy scores of 49.89%, 82.52%, and 84.98%, respectively. Comparing the results shows E-CNN and VGG-19 as the best performing models.

Furthermore, the analysis of the results reveals an interesting observation, in considering the two best performing models E-CNN and VGG19. Despite the fact that VGG19 model having an average accuracy score of 84.98% which is higher than the E-CNN model (83.87%), the E-CNN model performs better in generalization of malware classification based on malware families or types, of the 26 malware families or classes considered the E-CNN model outperformed by 14 malware classes compared to 11 malware classes by the VGG-19 model, while the remaining malware class was the same in both models. Figure 4.13 depicts this illustration

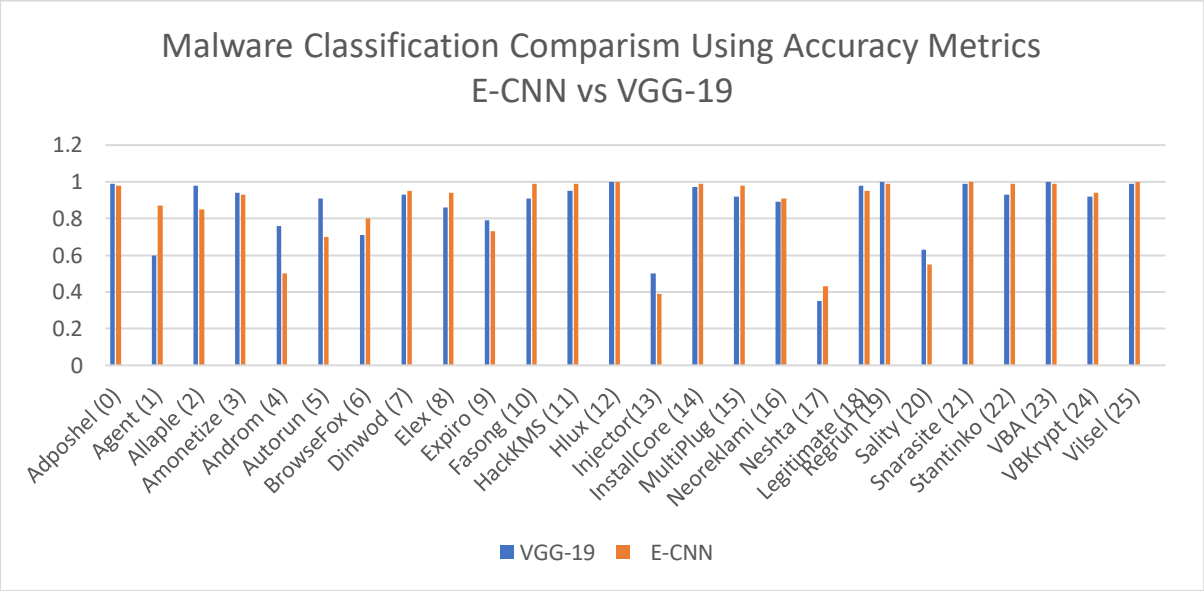


Figure 4.13: Figure showing the comparison between VGG-19 and E-CNN for malware classification

4.3.2 BENCHMARK DATASET EXPERIMENT AND EVALUATION

MALIMG DATASET

The developed E-CNN model was tested further using Malimg, a popular and publicly available malware dataset created in 2013. This is because the dataset is widely used by researchers to assess the performance of both machine and deep learning models for malware detection and classification. The dataset consists of malware samples which are divided into 25 families. As a

result, the Malimg dataset, is used as a benchmark performance measurement for the E-CNN model. Furthermore, the developed E-CNN architecture model is used to compare malware detection and classification models developed by other researchers.

The result of the E-CNN model experiment is evaluated using accuracy metrics, this indicates whether or not malware samples are correctly labelled. The model achieved an accuracy score of 98.88%. Thus, this reveals that this model is highly effective for malware classification of visual samples without using any data augmentation or data balancing method to enhance classification performance. This performance analysis is presented detail in the accuracy and loss graph curve, with the model trained in 16 epochs as shown in Figure 4.14 and Figure 4.15. Also, E-CNN model is evaluated in terms of confusion matrix this is shown in Figure 4.16.

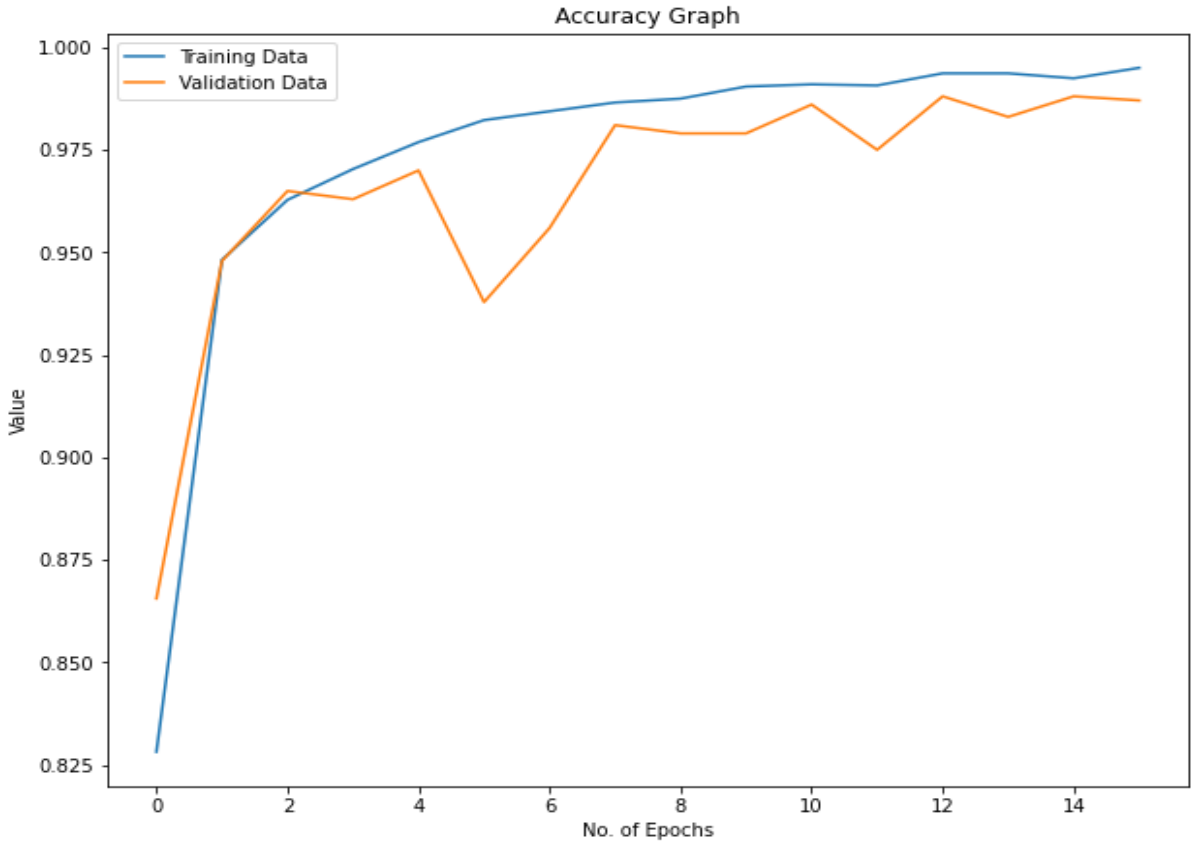


Figure 4.14: Accuracy graph showing the E-CNN model performance on Malimg dataset

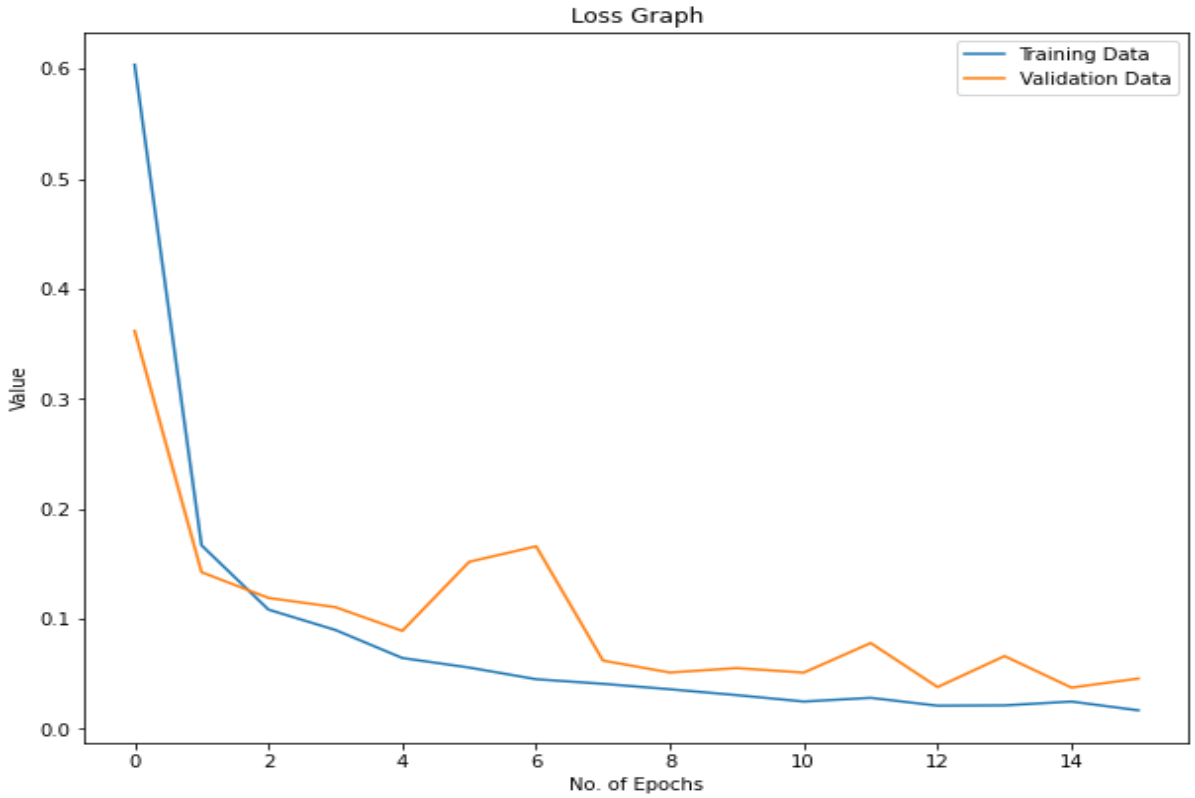


Figure 4.15: Loss graph showing E-CNN model performance on Malimg dataset

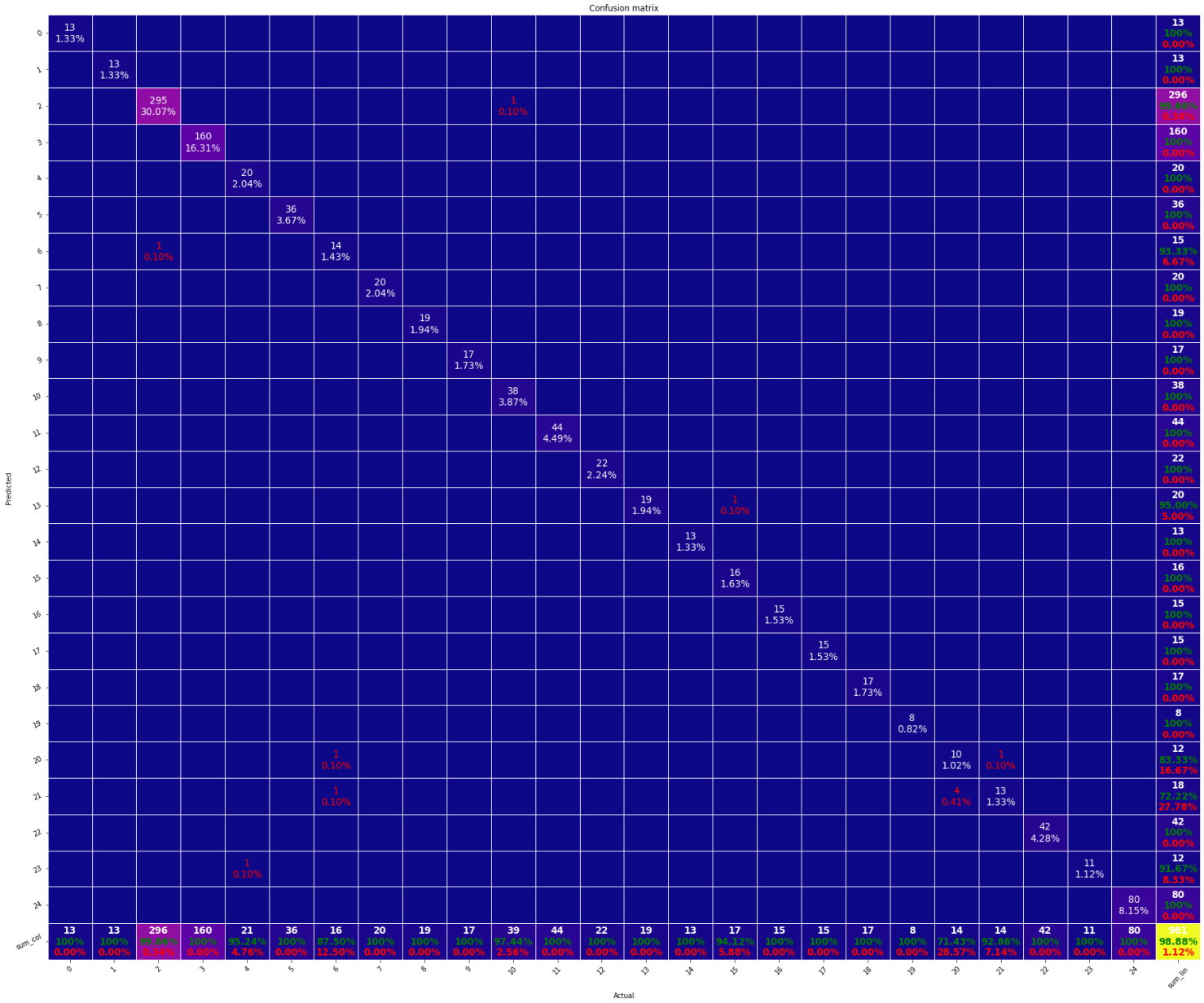


Figure 4.16: showing confusion matrix of malware classification of E-CNN architecture on Malimg dataset.

BENCHMARK COMPARISM WITH OTHER CITED WORKS

In this section, we assess the E-CNN model's performance using the malimg dataset. We further analyze the experiment's outcomes by comparing the model's results with those of other studies that also utilized the malimg dataset, measuring accuracy metrics. The results are presented in the Table 4.3

Table 4.3: Table comparing the proposed model with other models

s\n	Model Name	Authors	Accuracy (%)
1	EEMDS: Efficient and Effective Malware Detection System with Hybrid Model based on XceptionCNN and LightGBM Algorithm	(Onoja et al., 2022)	97.40 %
2	Attention-Based Cross-Modal CNN Using Non-Disassembled Files for Malware Classification	(Kim et al., 2023)	98.72%
3	Robust Malware Family Classification Using Effective Features and Classifiers	(Hammad et al., 2022)	96.64%
4	Malware classification through image processing with a convolutional neural network	(Marin et al., 2022)	98.70%
5	**This study**		**98.88%**

CHAPTER FIVE

5.0 SUMMARY

In this research paper, one of the objective of the research work was to develop a malware detection and classification model using CNN architecture, this was achieved using the malevis dataset which comprises of malware image samples of 25 distinct malware classes. The E-CNN model was able to detect and classify different malware samples into different classes having been able to identify unique structure of malware samples represented as images. Furthermore, the performance of the E-CNN model was shown to be very effect when the newly developed model was further tested by comparing it with estblished pre-trained CNN architectures, namely VGG16, VGG19, ResNet50 using the same malevis dataset. Subseqently, the proposed E-CNN model was applied on the popular Malimg dataset. Where the model achieved high accuracy in multi-classification of different malware image samples. Furthermore, the E-CNN model's result was also compared with results of other cited research work with the Malimg dataset being the benchmark dataset. The result showed E-CNN had a better performance when compared with some previous cited research work. Our findings indicate that the E-CNN model outperformed some of the previously cited research works, showcasing its effectiveness in the field of malware detection and classification.

5.1 CHALLENGES AND ISSUES

From the review of the recent studies, as well as the experiment conducted several noteworthy insights have emerged, shedding light on the challenges inherent in the development of malware classification models. These identified challenges subsequently give rise to a multitude of issues within the realm of image-based malware classification. The crux of achieving successful classification lies in the dual qualities of consistency and effectiveness in the classifier's performance. Constructing such a classifier necessitates a comprehensive consideration of all the intricacies and obstacles that are entailed, which are as follows

Datasets Used

The datasets commonly employed by most researchers for malware classification within the reviewed literature, includes well-known datasets such as "Malimg" and the "Microsoft Malware Dataset 2015" although popular among researchers they could exhibit limitations in their effectiveness when utilized for developing models that can classify newer malwares. This shortcoming arises from the fact that constructing a model based on outdated malware samples

renders it ineffectual against contemporary malware threats. Consequently, the persistently evolving tactics of malware authors, who predominantly utilize modern malware samples, result in the classifiers' inability to accurately categorize these new malicious entities.

Furthermore, these datasets suffer from a limitation of diverse malware samples. This scarcity poses a significant challenge, particularly for CNN models which thrive on substantial data volumes to achieve robust training and model development. Insufficient samples within a dataset can induce overfitting in the models, wherein they become overly specialized to the limited data and consequently fail to effectively identify novel malware instances.

Performance Computation Measures

The computational costs associated with most of these models frequently exhibit a high degree of resource consumption, sometimes with unclear correlations to their performance in malware classification. This ambiguity spans multiple facets, encompassing the definition of performance metrics, the time required for training and testing, and the intricacies of translating malware binaries into colour images. Furthermore, approaches aimed at mitigating data imbalance issues during the classification of malware families, as well as efforts to condense the feature vector's dimensions, assume noteworthy importance. This is particularly significant since the size of the feature vector significantly impacts the overall efficiency attainable by these models.

Data enhancement/Data augmentation

The use of data enhancement and augmentation on datasets although can improve model's performance, however researchers need to exercise measure of caution or balance when employing these methods so as to prevent overfitting hence models memorize only training data and cannot adapt to actual data problems

5.2 CONTRIBUTION TO KNOWLEDGE

This research work highlights a few noteworthy mentions which are as follows

- The E-CNN model demonstrated a high level of cross-dataset generalization, as evident in our experiment. We developed the E-CNN on a recent dataset and assessed its performance using a well-established, albeit outdated, benchmark dataset. Notably, it achieved a remarkable level of accuracy. This adaptability is crucial in the ever-changing field of malware detection, as the model possesses the capability to detect previously unseen malware variants.

- During this research, most researchers applied generalized models through transfer learning to develop malware classification models. However, these models are susceptible to domain mismatches. In contrast, the E-CNN model can serve as a robust foundation for the development of more effective malware classification models

5.3 CONCLUSION AND FUTURE WORKS

The threat of malwares on information systems, computers is continuously evolving and changing hence the need for malware researchers to continually develop new methods or techniques in addressing this problem. In this research, a E-CNN model is proposed which can classify different malware types effectively and efficiently.

This research work aims to contribute to the advancement of developing models for detection and classification of malwares, using CNN. In the domain of malware visualization using CNN architecture, CNN aids analysts in identifying crucial image patterns. Based on the findings, it is clear that pursuing malware visualization in conjunction with the CNN approach can yield a more intelligent framework. This framework promises to enhance accuracy, efficiency, and overall performance, all of which are vital in the ever-evolving landscape of malware threats.

For future studies, several recommendations are worth noting:

1. **Dataset Choice**: Utilize a large, up-to-date malware dataset containing recent malware samples. This is crucial for assessing and validating performance measures effectively.
2. **Image Conversion**: Explore more efficient techniques for converting malware binaries into colour images, considering variations in image sizes across different datasets.
3. **Model Development**: While transfer learning has its merits, consider focusing on the development of novel models. This approach can help mitigate errors stemming from domain mismatch, where a model trained in one domain is applied to a different one.
4. **Feature Vector Dimensionality**: Reduce the dimensionality of the feature vector to enhance model efficiency.
5. **Data Imbalance**: Implement newer methods or techniques to address data imbalance problems effectively.

In the course of this research, significant knowledge gaps have been revealed, major challenges have been identified, also highlighted are open issues that will serve as valuable guides for future research endeavours.

REFERENCES

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., & Arshad, H. J. H. (2018). State-of-the-art in artificial neural network applications: A survey. *4*(11), e00938.

Agrawal, R., & Khan, L. (2021). AN EXPERIENCE IN ENHANCING MACHINE LEARNING CLASSIFIER AGAINST LOW-ENTROPY PACKED MALWARES.

Ahmed, M., Afreen, N., Ahmed, M., Sameer, M., & Ahamed, J. (2023). An inception V3 approach for malware classification using machine learning and transfer learning. *International Journal of Intelligent Networks*, *4*, 11-18.

Al-Humaidan, N. A., & Prince, M. (2021). A classification of Arab ethnicity based on face image using deep learning approach. *IEEE Access*, *9*, 50755-50766.

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. 2017 International Conference on Engineering and Technology (ICET),

Alshamrani, S. S. (2022). Digital Forensics for Malware Classification: An Approach for Binary Code to Pixel Vector Transition. *Computational Intelligence and Neuroscience*.

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of big Data*, *8*(1), 1-74.

Anaconda. (2023). *Anaconda* Anaconda.com. Retrieved 12- Jan- 2023 from www.anaconda.com

Asam, M., Khan, S. H., Jamal, T., Zahoora, U., & Khan, A. (2021). Malware Classification Using Deep Boosted Learning. *arXiv preprint arXiv:2107.04008*.

Awan, M. J., Masood, O. A., Mohammed, M. A., Yasin, A., Zain, A. M., Damaševičius, R., & Abdulkareem, K. H. (2021). Image-Based Malware Classification Using VGG19 Network and Spatial Convolutional Attention. *Electronics*, *10*(19), 2444.

Bae, S. I., Lee, G. B., & Im, E. G. (2020). Ransomware detection using machine learning algorithms. *Concurrency and Computation: Practice and Experience*, *32*(18), e5422.

Bensaoud, A., Abudawaood, N., & Kalita, J. (2020). Classifying malware images with convolutional neural network models. *International Journal of Network Security*, *22*(6), 1022-1031.

Brownlee, J. (2019). *Deep learning for Computer Vision*.

Cakir, B., & Dogdu, E. (2018). Malware classification using deep learning methods. Proceedings of the ACMSE 2018 Conference,

Carletti, V., Greco, A., Saggese, A., & Vento, M. (2021). Robustness evaluation of convolutional neural networks for malware classification. *ITASEC 2021*

Italian Conference on Cybersecurity 2021, *2940*, 414-423.

Chauhan, R., Ghanshala, K. K., & Joshi, R. (2018). Convolutional neural network (CNN) for image detection and recognition. 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC),

Deng, X., & Mirkovic, J. (2022). Polymorphic Malware Behavior Through Network Trace Analysis. 2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS),

Dwivedi P, & Sharan H. (2021). Analysis and Detection of Evolutionary Malware: A. *International Journal of Computer Applications*, *975*, 8887.

El-Shafai, W., Almomani, I., & AlKhayer, A. (2021). Visualized malware multi-classification framework using fine-tuned CNN-based transfer learning models. *Applied Sciences*, *11*(14), 6446.

Ghosh, A. (2021). An overview article on 600% increase in Cyber Attack in 2021.

Gibert, D., Mateu, C., & Planes, J. (2020). The rise of machine learning for detection and classification of malware: Research developments, trends and challenges. *Journal of Network and Computer Applications*, *153*, 102526.

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1). MIT press Cambridge.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., & Cai, J. (2018). Recent advances in convolutional neural networks. *Pattern recognition*, *77*, 354-377.

Guo, T., Dong, J., Li, H., & Gao, Y. (2017). Simple convolutional neural network on image classification. 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA),

Hammad, B. T., Jamil, N., Ahmed, I. T., Zain, Z. M., & Basheer, S. (2022). Robust Malware Family Classification Using Effective Features and Classifiers. *Applied Sciences*, *12*(15), 7877.

Kalash, M., Rochan, M., Mohammed, N., Bruce, N. D., Wang, Y., & Iqbal, F. (2018). Malware classification with deep convolutional neural networks. 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS),

Keahey, T. A. (2013). Using visualization to understand big data. *IBM Business Analytics Advanced Visualisation*, *16*.

Khan, R. U., Zhang, X., & Kumar, R. (2019). Analysis of ResNet and GoogleNet models for malware detection. *Journal of Computer Virology and Hacking Techniques*, *15*, 29-37.

Kim, J., Paik, J.-Y., & Cho, E.-S. (2023). Attention-Based Cross-Modal CNN Using Non-Disassembled Files for Malware Classification. *IEEE Access*, *11*, 22889-22903.

Kolosnjaji, B., Zarras, A., Webster, G., & Eckert, C. (2016). Deep Learning for Classification of Malware System Call Sequences. In B. H. Kang & Q. Bai, *AI 2016: Advances in Artificial Intelligence* Cham.

Kumar, S. (2021). MCFT-CNN: Malware classification with fine-tune convolution neural networks using traditional and transfer learning in internet of things. *Future Generation Computer Systems*, *125*, 334-351.

Le, Q., Boydell, O., Mac Namee, B., & Scanlon, M. (2018). Deep learning at the shallow end: Malware classification for non-domain experts. *Digital Investigation*, *26*, S118-S126.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436-444.

Lin, W.-C., & Yeh, Y.-R. (2022). Efficient Malware Classification by Binary Sequences with One-Dimensional Convolutional Neural Networks. *Mathematics*, *10*(4), 608.

Liu, J., Zeng, Y., Shi, J., Yang, Y., Wang, R., & He, L. (2019). Maldetect: a structure of encrypted malware traffic detection. *CMC-COMPUTERS MATERIALS & CONTINUA*, *60*(2), 721-739.

Lo, W. W., Yang, X., & Wang, Y. (2019). An xception convolutional neural network for malware classification with transfer learning. 2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS),

Loukadakis, M., Cano, J., & O'Boyle, M. (2018). Accelerating deep neural networks on low power heterogeneous architectures.

Marin, D., Orozco-Rosas, U., & Picos, K. (2022). Malware classification through image processing with a convolutional neural network. Optics and Photonics for Information Processing XVI,

Matplotlib. (2023). *Matplotlib: Visualization with python*. Matplotlib. Retrieved 5 Jan 2023 from https://matplotlib.org/

Meng, X., Shan, Z., Liu, F., Zhao, B., Han, J., Wang, H., & Wang, J. (2017). MCSMGS: malware classification model based on deep learning. 2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC),

Moussas, V., & Andreatos, A. (2021). Malware detection based on code visualization and two-level classification. *Information*, *12*(3), 118.

Mumtaz, Z., Afzal, M., Iqbal, W., Aman, W., & Iltaf, N. (2021). Enhanced Metamorphic Techniques-A Case Study Against Havex Malware. *IEEE Access*, *9*, 112069-112080.

Nataraj, L., Karthikeyan, S., Jacob, G., & Manjunath, B. S. (2011). Malware images: visualization and automatic classification. Proceedings of the 8th international symposium on visualization for cyber security,

Nguyen, T.-H., Nguyen, T.-N., & Ngo, B.-V. (2022). A VGG-19 Model with Transfer Learning and Image Segmentation for Classification of Tomato Leaf Disease. *AgriEngineering*, *4*(4), 871-887.

Nisa, M., Shah, J. H., Kanwal, S., Raza, M., Khan, M. A., Damaševičius, R., & Blažauskas, T. (2020). Hybrid malware classification method using segmentation-based fractal texture analysis and deep convolution neural network features. *Applied Sciences*, *10*(14), 4966.

O'Shaughnessy, S., & Sheridan, S. (2022). Image-based malware classification hybrid framework based on space-filling curves. *Computers & Security*, *116*, 102660.

Onoja, M., Jegede, A., Blamah, N., Abimbola, O. V., & Omotehinwa, T. O. (2022). EEMDS: Efficient and Effective Malware Detection System with Hybrid Model based on XceptionCNN and LightGBM Algorithm. *Journal of Computing and Social Informatics*, *1*(2), 42-57.

Pandas. (2023). *Pandas - Python Data Analysis Library*. Pandas. Retrieved Jan 13 from https://pandas.pydata.org

Prajapati, P., & Stamp, M. (2021). An empirical analysis of image-based learning techniques for malware classification. *Malware analysis using artificial intelligence and deep learning*, 411-435.

Rot, A., & Olszewski, B. (2017). Advanced Persistent Threats Attacks in Cyberspace. Threats, Vulnerabilities, Methods of Protection. FedCSIS (Position Papers),

Sahay, S. K., Sharma, A., & Rathore, H. (2020). Evolution of malware and its detection techniques. In *Information and Communication Technology for Sustainable Development* (pp. 139-150). Springer.

Samek, W., Binder, A., Montavon, G., Lapuschkin, S., & Müller, K.-R. (2016). Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, *28*(11), 2660-2673.

Schofield, M., Alicioglu, G., Binaco, R., Turner, P., Thatcher, C., Lam, A., & Sun, B. (2021). Convolutional neural network for malware classification based on API call sequence. Proceedings of the 8th International Conference on Artificial Intelligence and Applications (AIAP 2021),

Tahir, R. (2018). A study on malware and malware detection techniques. *International Journal of Education and Management Engineering*, *8*(2), 20.

Wajahat, A., Imran, A., Latif, J., Nazir, A., & Bilal, A. (2019). A Novel Approach of Unprivileged Keylogger Detection. 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET),

Yoo, S., Kim, S., Kim, S., & Kang, B. B. (2021). AI-HydRa: Advanced hybrid approach using random forest and deep learning for malware classification. *Information Sciences*, *546*, 420-435.

Yu, W., Yalin, Y., & Haodan, R. (2019). Research on the technology of trojan horse detection. 2019 12th International Conference on Intelligent Computation Technology and Automation (ICICTA),

Yuan, B., Wang, J., Liu, D., Guo, W., Wu, P., & Bao, X. (2020). Byte-level malware classification based on markov images and deep learning. *Computers & Security*, *92*, 101740.

Yuan, B., Wang, J., Liu, D., Guo, W., Wu, P., Bao, X. J. C., & Security. (2020). Byte-level malware classification based on markov images and deep learning. *92*, 101740.

Zhang, Z. (2018). Artificial neural network. In *Multivariate time series analysis in climate and environmental research* (pp. 1-35). Springer.

Dissertation

by Acetel ACETEL

Submission date: 09-Feb-2024 02:02PM (UTC+0100)
Submission ID: 2268370628
File name: ACE21140004_PhD_Thesis_Manuscript.pdf (3.08M)
Word count: 18168
Character count: 113687

An Intelligent Sign Language Recognition System for the Deaf and Hard of Hearing

By

Mustapha Deji DERE

ACE21140004

National Open University of Nigeria

December 2023

An Intelligent Sign Language Recognition System for the Deaf and Hard of Hearing

By

Mustapha Deji DERE

ACE21140004

A Thesis Submitted in Partial Fulfilment of the Requirements for the Award of the Degree of Doctor of Philosophy (Ph.D.) in Artificial Intelligence
At the Africa Centre of Excellence on Technology Enhanced Learning
National Open University of Nigeria

Declaration

I, **Mustapha Deji DERE,** hereby declare that the project work entitled

An Intelligent Sign Language Recognition System for the Deaf and Hard of Hearing is a record of an original work done by me, as a result of my research effort carried out in the Africa Centre of Excellence on Technology Enhanced Learning, National Open University of Nigeria under the supervision of

Prof. Aliyu Rufai Yauri and **Dr. Adewale Adesina**.

8th December 2023

_____

Student's Signature & Date

Certification / Approval

This is to certify that this study was carried out by **Mustapha Deji DERE** Matric Number **ACE21140004** in the **Department of Artificial Intelligence**, **Africa Centre of Excellence on Technology Enhanced Learning**, National Open University of Nigeria, under my supervision.

.....	.....
Chief Supervisor	Sign & Date
.....	.....
Supervisor	Sign & Date
.....	.....
Centre Director	Sign & Date
.....	.....
Program Co-ordinator	Sign & Date
.....	.....
External Examiner	Sign & Date

Dedication

This thesis is dedicated to the entire DERE family, and the people listed in Appendix IX for their unending love, support, and encouragement throughout my academic journey. Their belief in me has been the guiding force in completing this thesis.

Acknowledgements

In the name of Allah, the Most Gracious, the Most Merciful. All praises and thanks belong to Allah alone, without any partners. May the blessings and salutations of Allah be upon the seal of all prophets, Muhammed, peace be upon him. Reflecting on my journey, it is astounding to see how far I have come. From my humble beginning with disappointing examination results to where I stand today as a Ph.D. candidate, only by the grace of Allah, and the influence of incredible individuals, was I able to achieve this tremendous accomplishment.

I would like to extend my deepest appreciation to my esteemed supervisor, Prof. Aliyu Rufai Yauri, and Dr. Adewale Adesina for entrusting me with the pursuit of my Ph.D. in a subject that is particularly close to my heart. Without their unwavering guidance and support, my journey through this challenging academic pursuit would have been futile. I am eternally grateful for their encouragement, and commitment to my success. Their mentorship has been instrumental in helping me achieve my academic dreams, and I will always treasure their invaluable contribution to my academic and professional development.

I would like to express my heartfelt gratitude to the numerous individuals in my life who have contributed in no small measure to my success. I am unable to thank all of you exhaustively, but I would like to extend a special thanks to my loving wife and kids for their patience and unwavering support, and to my parents - Mr. and Mrs. Rasheed Dere, Mr. and Mrs. Bisi Dere, Mr. and Mrs. Biodun Dere, and Mr. and Mrs. Taiwo Dere - among others (see Appendix IX). Additionally, I would like to acknowledge the unwavering support of my dear friends - Arch. Ibrahim Kalejaiye, Engr. Ahmed Gbigbadua, and Engr. Kehinde Kamil. Your friendship has been a source of strength and motivation, and I am deeply appreciative of your encouraging words and unwavering support throughout my journey.

Table of Contents

List of Figures

List of Tables

Abbreviations

AI	Artificial Intelligence
AKSL	Akure Sign Language
ANN	Artificial Neural Network
API	Application Programming Interface
ASL	American Sign Language
BLE	Bluetooth
BLEU	Bilingual Evaluation Understudy Score
CNN	Convolution Neural Network
CV	Computer Vision
DCNN	Deep Convolution Neural Network
DHH	Deaf and hard of hearing
DNN	Deep Neural Network
EMG	Electromyography
FF	Feed-Forward
FN	False Negative
FP	False Positive
GPU	Graphic Processing Unit
GRU	Gated Recurrent Network
HMM	Hidden Markov Model
HSL	Hausa Sign Language
IMU	Inertia Measurement Unit

INSL	Indigenous Nigerian Sign Language
JW	Jehovah Witness
KNN	K-Nearest Neighbor
LN	Layer Normalization
LRL	Low-Resource Language
LSTM	Long Short Time Memory
MHA	Multi-Head Attention
ML	Machine Learning
NLLB	No Language Left Behind
NLP	Natural Language Processing
NSL	Nigeria Sign Language
PCA	Principal Component Analysis
RNN	Recurrent Neural Network
SLR	Sign Language Recognition
SLRT	Sign Language Recognition and Translation
SLT	Sign Language Translation
SOTA	State of the Art Accuracy
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
VR	Virtual Reality
WFD	World Federation of the Deaf
YOLO	You Only Look Once

Appendices

Abstract

Individuals with hearing and speech impairments often encounter significant challenges when it comes to integrating into various aspects of society, particularly in terms of job accessibility, media, and education. This limitation in communication abilities can impede their ability to swiftly access medical assistance during emergencies. While research and technologies have been developed to address the specific needs of the deaf and hard of hearing (DHH) community, there has been a lack of focus on edge deployment and sign language recognition with simultaneous translation to low-resource languages. In response to this, a methodical approach called SmartCall and SmartCom has been proposed to tackle these challenges. SmartCall is an edge resource-constrained solution that utilizes AI to enable individuals with speech impairments to communicate during medical emergencies. It employs a 1D-convolution neural network (CNN) model to extract features from an inertial measurement unit (IMU) on-board, which is trained to recognize selected American Sign Language (ASL) medical emergency words in real-time on a low-cost embedded device. On the other hand, SmartCom is an end-to-end framework designed to translate ASL to three indigenous languages in Nigeria - Hausa, Ibo, and Yoruba. This system utilizes a pre-trained CNN model and a Transformer-based model for ASL-to-text generation, supported by a neural language model called "No Language Left Behind" (NLLB) to translate the generated text from English to the target language. During validation, SmartCall demonstrated an offline accuracy of 91.2% and an average on-device accuracy of over 90% for the 8-bit optimized model. The performance of the ASL-to-text model was evaluated using BLEU scores, resulting in BLEU-1, BLEU-2, BLEU-3, and BLEU-4 scores of 18.55, 8.99, 4.99, and 2.96 respectively. Qualitative analysis indicated that participants were satisfied with both the ASL-to-Text and Text-to-Low-resource language models. Despite the

framework's limitations, such as the absence of testing with native low-resource language speakers with hearing and speech impairments to enhance its applicability, the study's findings represent significant progress in the quest for improving communication and promoting inclusivity for individuals with disabilities in various sociocultural contexts and education.

Keywords: American sign language (ASL), deep learning, low-resource language, tinyML.

CHAPTER ONE

INTRODUCTION

1.1. Background of the Study

Communication plays an important role in day-to-day human interaction, however, people living with deaf and hard of hearing (DHH) find it difficult to communicate using the traditional methods such as speech and hearing. Sign language serves as one of the means of communication used in the DHH community. It is a natural language which has its own unique grammar, semantics, facial expressions, and body language (Liddell 2003). In addition to sign language, writing is another common means of communication for the DHH community. Recent developments in technology have led to the creation of mobile and online applications specifically designed for DHH individuals. However, certain conditions such as medical distress make writing-based innovations impractical (Dere et al. 2023).

Sign languages are being adopted by numerous individuals all over the world, according to a survey carried out by the World Federation of the deaf (WFD), it was reported that around 70 million people are living with hearing impairment globally, with Africa having a higher rate (Mulwafu, Kuper, and Ensink 2016), it was reported in a research carried out in the North-West Cameroon that people around the age of 50 years and above are great risk (Ferrite et al. 2017).Various studies have accentuated the effect and factors contributing to hearing loss in Nigeria (Ahmed 2016; Okunade 2018; Wouters et al. 2020). Numerous individuals living with hearing loss are unable to receive the required support needed to achieve their full potentials (Dere et al. 2023)because of unavailability of early intervention and rehabilitation services in Nigeria (Ogunkeyede et al. 2019).

Hearing and speech impairments is a causative factor that hinders the individuals with DHH from accessing employment opportunities, social activities, and education (Asonye

2022; Asonye, Emma-Asonye, and Edward 2018). Asonye (2022) was of the opinion that there are two major solutions to resolving the existing gap among deaf children in Nigeria(Dere et al. 2023): the embracement and application of indigenous sign languages, and the execution of early detection and intervention programs. The aggravated part of the problem is the fact that there are at least 100 low-resource languages (LRL) spoken all over Nigeria without an officially recognized Nigerian Sign Language (NSL) yet(Dere et al. 2023). There are several indigenous sign languages in Nigeria such as Akure sign language (AKSL) (Orie 2013), Hausa sign language (HSL) (Hassan et al. 2017), and Bura sign language (Blench, Warren, and UBS 2006). However, the sign language commonly used in Nigerian schools for the deaf, known as NSL, is mainly based on American sign language (ASL) vocabulary (ASONYE and EDWARD 2022). Despite the existence of other sign languages across the globe (Abdul et al. 2021; Gupta and Kumar 2021; Khomami and Shamekhi 2021), ASL is by large the common language among the DHH community around the globe (Cheok, Omar, and Jaward 2019), and in Nigeria (ASONYE and EDWARD 2022).

The new wave in technology has played an important role in eliminating the communication barrier between DHH individuals and those with normal hearing. (Falvo, Scatalon, and Francine Barbosa 2020). Several sensor-based technologies have been applied to promote seamless communication among the DHH community. Electromyography (EMG) biosensor (Alwarfalli, Yousuf, and Ighneiwa 2021; Zhang et al. 2019), datagloves (Faisal et al. 2022; Wen et al. 2021), IMU (Deji Dere et al. 2022; Pereira-Montiel et al. 2022), stretchable sensors (Zhou et al. 2020), and cameras sensors (Barbhuiya, Karsh, and Jain 2021; Magdy et al. 2022; Sharma and Kumar 2021; Sharma and Singh 2021).Several sensor-based technology has been executed for tasks such as sign language recognition (SLR), sign

language translation (SLT), sign language retrieval, and sign language recognition and translation (SLRT). Regardless of how advantageous the technologies are, very limited number of studies have proposed a comprehensive framework for ASL recognition and consequent translation to low-resource languages in Nigeria (Dere et al. 2023). Developing a framework for an end-to-end SLRT framework from ASL to low-resource languages could contribute immensely inclusivity and open more employment opportunities in the education sector for people living with DHH (Dere et al. 2023).

Active sensor-based technologies are usually dependent on their users which eventually reduce their implementation in several contexts. Conversely, cameras offer a greater scope of applications and expanded audience, although their performance has been influenced by some factors such as restricted field of view, lighting conditions, and other environmental variables. Despite these hindrances, camera sensors are most effective for tasks like training and social media reporting, especially in situations involving DHH individuals (Dere et al. 2023).

To achieve efficient recognition of patterns from sensors for sign language communication, several pattern recognition algorithms have been proposed. A hidden Markov model (HMM) was implemented by (Liu, Jiang, and Gowda 2020) for sign language communication recognition. A support vector machine (SVM) was proposed by (Quinn and Olszewska 2019) for efficient sign language pattern recognition. Achieving high accuracy in conventional machine learning method for sign language recognition requires domain-specific feature extraction.

An advantageous aspect of deep learning is its ability to automate the feature extraction process without domain knowledge, even though this may come at the cost of increased

computational and energetic requirements (Dodge et al. 2022; Patterson et al. 2021). While deploying a DNN to an embedded device presents a potential solution to address the energy consumption and latency concerns, achieving real-time DNN deployment on an embedded device remains a non-trivial task (Han and Siebert 2022; Janapa Reddi et al. 2022; Shafique et al. 2021). Moreso, Convolutional Neural Networks (CNN) are frequently employed in computer vision tasks, while Recurrent Neural Networks (RNNs) are often utilized for translation tasks. Recently, Transformer algorithms have successfully outperformed CNNs and RNNs in various computer vision (Han et al. 2023; Khan et al. 2022) and natural language processing tasks (NLP) (Wolf et al. 2020) respectively. Therefore, it is necessary to perform the appropriate pattern recognition algorithm, together with the specific type of neural network hyperparameters, to ensure optimal performance when developing SLR and SLRT frameworks (Dere et al. 2023).

1.2. Statement of the Problem

Recent technological advancements have enabled DHH solutions chiefly through artificial intelligence (AI) powered innovations. Despite these innovations, DHH individuals still face limitations in situations where writing-based communication is impractical, and the cost of wearable sensors is expensive, resulting in a lack of support for education, employment opportunities, and during medical distress. The problem is further aggravated in Nigeria due to the lack of official indigenous Nigerian sign language (INSL). This thesis aims to fill the gap by introducing a tiny DNN that facilitates communication for DHH individuals in distressing situations. Furthermore, the research presents an end-to-end framework for translating America sign language to indigenous Nigerian languages, specifically for educational purposes.

Conventional machine learning algorithms used for pattern recognition often require manual feature extraction, which may not be the best approach for certain applications. In contrast, deep learning algorithms provide automatic feature extraction with high performance results. However, one limitation of using deep learning is the latency and energy consumption associated with training and deployment. Selecting the appropriate architecture and hyperparameters can be challenging and may result in suboptimal performance if not chosen correctly. This thesis addresses these issues by proposing the deployment of a tiny deep neural network (TinyML) on an edge device for sign word recognition. Additionally, the thesis investigates different variation of a Transformer-based model to achieve effective recognition of ASL and sequential translation to LRL in Nigeria.

1.2.1. Research Questions

The questions formulated from the statement of problems, aim of the study, and specific objectives are as follows:

- How efficient and cost effective is the AI-based system for communication during medical distress for DHH individuals?
- What is the performance of the model offline and on-device?
- How effective are Transformer models for ASL sign recognition from video data and simultaneous generation of English text?
- What is the performance of the developed ASL-to-text model?
- How effective is the adapted translation model for English to three low-resource language such as Hausa, Ibo, Yoruba in Nigeria in generating comprehensible text?
- How well do native speakers comprehend the generated text?

1.3. Aim of the Study

The purpose of this study is to explore the potential of leveraging AI-based technologies to address challenges faced by DHH individuals during medical emergencies and when delivering educational instructional content as an instructor or educator.

1.4. Specific Objectives

The specific objectives are as follows:

- To develop an effective AI-based system for communication during medical distress.
- To deploy the model to an edge device and evaluate the performance of offline and on-device.
- To implement Transformer models for ASL sign recognition from video data and simultaneous generation of English text.
- To quantitatively access the performance of the ASL-to-Text model.
- To examine the comprehension of the generated America sign language-to-Text and America sign language-to-low-resource-language models.
- To assess the quality of the translated text and determine the user's level of satisfaction.

1.5. Scope of the Study

The scope of this thesis involved using both private and public data to evaluate the performance of the proposed solutions. Private data is used with the IMU-based ASL recognition algorithm as it pertains to the deployment phenomenon. Public data was used to ensure generalization in the ASL-to-text model that uses Transformers as a data-involved architecture.

1.6. Significance of the Study

The significance of this thesis lies in the potential for leveraging AI-based technologies to enhance communication and provide better access to emergency medical services and job opportunities in the educational sector for the DHH community. The thesis findings could contribute to the development of more accessible and effective communication technologies for the DHH community, potentially opening job opportunities in the education and media sectors of the Nigerian economy.

1.7. Definition of Terms

- American Sign Language (ASL): This is a visual language leveraged by DHH individuals to communicate.
- Deep Learning: This is a subset of machine learning algorithm that uses backpropagation for pattern recognition without manual feature extraction.
- Low-Resource Languages (LRL): This is a language that has relatively little data available for training natural language processing (NLP) systems.
- TinyML: This is a new domain focusing on deployment of DNN to embedded devices.
- Transformers: This is a DNN architecture based on attention mechanism typically used in NLP domain.

1.8. Organization of the Thesis

The thesis is a combination of two solutions towards enhancing opportunities and building communication gaps in the DHH community by leveraging AI-based technologies. In the following sections, we provide a detailed review of relevant literature in CHAPTER TWO. CHAPTER THREE covers the methodology, data used, details on training the DNN, and the experimental setup. We present the experimental results and discussion in CHAPTER FOUR before concluding our research in CHAPTER 5. Additionally, we provide recommendations and future directions for further study.

CHAPTER TWO

LITERATURE REVIEW

2.1. Preamble

The literature review section has been divided into several parts which include a theoretical framework, a review of relevant literature, a review of related works, and a summary of these works. The theoretical framework section will delve into the fundamentals of machine learning, deep learning, computer vision (CV), and NLP. The relevant literature review section encompasses a review of state-of-the-art architectures in both CV and NLP domains. Moreover, the review of related works section is subdivided into four parts, which are sensing modalities that are categorized as camera-based and wearable-based approaches, the pattern recognition algorithms employed SLR, an in-depth approach for INSL, and a comprehensive review of recent research conducted in SLRT. Lastly, a succinct summary of all the relevant works is presented.

2.2. Theoretical Framework

2.2.1. Machine Learning

Machine learning (ML) is a type of AI that focuses on using data and algorithms to allow machines to learn from the data and improve their performance on various tasks without being explicitly programmed to do so (Chollet 2017). ML algorithms build a model based on training data to make predictions or decisions. These algorithms are used in a wide range of applications, including medicine, email filtering, speech recognition, agriculture, and CV. ML is closely related to the fields of computational statistics and data mining. The goal of ML is to achieve generalization, where the algorithm can perform well on unseen data. It is also important to note that while machine learning and statistics use similar methods, their goals differ. Some examples of conventional machine learning algorithms are KNN, Naïve Bayes, Linear Regression, and SVM amongst others. A significant limitation of the conventional machine learning method is the need for domain-specific feature extraction. Typically, the deep learning approach is a subset of machine learning, however, in this thesis deep learning is used as a distinct method which involves the use of backpropagation.

2.2.2. Deep Learning

Deep learning is a specific type of machine learning that uses computational graphs in the form of deep layers to automatically extract features from raw data (Courville 2016). The deep learning approach was first introduced in the form of multi-layered perceptron which is also referred to as artificial neural network (ANN). Neural networks learn patterns and features in large datasets. Deep learning algorithms can autonomously discover complex representations or abstractions of the input data, making them extremely efficient in analyzing images, videos, audio, and natural language. Deep learning has been successful in several applications, including object recognition, speech recognition, and NLP. Overall, deep learning is a powerful tool in AI

that allows computers to learn from and analyze complex data in a way that was not possible before. A significant drawback of deep learning is the need for big data to learn features without overfitting. Additionally, the cost of training and deploying DNN are a constraint for embedded devices.

2.2.3. Computer Vision

Computer vision is a field of AI focused on enabling computers to obtain information and understand the content of digital images or video, derived from machines or a set of webcams. The field of CV involves developing algorithms and techniques for machines to see and interpret the visual world and extract useful information or insights from the data (Forsyth and Ponce 2002). Conventional machine learning algorithms such as SVMs (Chandra and Bedi 2021) and Random Forests (Horning 2010) are two popular conventional machine learning algorithms widely used in CV. However, Deep learning algorithms have recently shown significant progress in different computer vision tasks, especially image classification and object detection (Chai et al. 2021).

2.2.4. Natural Language Processing

Natural language processing is a field of AI that focuses on enabling computers to understand, interpret and generate human language. NLP is used to analyze and process human language at various levels, including semantics, and pragmatics. Traditional machine learning algorithms have been used extensively in NLP. For instance, these techniques are used for text classification (Occhipinti, Rogers, and Angione 2022), text generation , and language translation.

Deep learning algorithms like RNNs, Long Short-Term Memory (LSTM), and Transformer models, have significantly improved the state-of-the-art results in NLP applications, such as machine translation (Kahlon and Singh 2021), language generation (Iqbal and Qureshi

2022), and text summarization. These models can learn complex linguistic patterns in text data and can model the long-term dependencies between words in sequences.

2.3. Review of Relevant Literature

2.3.1. State-of-the Art Architecture in CV

The success of AlexNet (Krizhevsky, Sutskever, and Hinton 2017) on the ImageNet dataset revolutionized the field of AI. AlexNet made significant contributions to the field of AI, specifically in the area of computer vision. Before the development of AlexNet, the state-of-the-art models for image classification used shallow machine learning models and were not able to perform effectively on large datasets. However, AlexNet demonstrated that with sufficient computation tools, dataset and DNN, deep learning algorithms can yield better performance with raw data without feature extraction.

CNN has been the gold-standard in computer vision using a deep learning approach. particularly the Residual Network (ResNet). (He et al. 2016) introduces ResNet to address the problem of vanishing gradients. One of its key features is the use of skip or residual connections, which enable information to be directly passed from one layer to another, bypassing the intermediate layers. ResNet's architecture allows for much deeper networks to be trained without the degradation of accuracy that typically occurs with deep neural networks. A limitation of CNN in CV task is the inability to generalize global concept in long dependencies.

Vision Transformers (ViTs) were introduced to solve this problem (Dosovitskiy et al. 2020). ViTs are a type of neural network architecture that use self-attention mechanisms to process images, allowing them to outperform traditional CNNs on various CV tasks. ViTs break down the input image into a grid of small patches and linearly project each patch into a lower-dimensional embedding space. The embedding vectors of each patch are then fed through an

attention mechanism which allows the model to focus on specific parts of the image and assign different weights to different patches based on their relevance to the task at hand. The resulting weighted embeddings are then fed through a feedforward neural network to produce the final output.

2.3.2. State-of-the Art Architecture in NLP

One of the key developments in NLP architecture in recent years is the use of the Transformer model, which was introduced by (Vaswani et al. 2017) and has since become the model of choice for various NLP tasks. Transformers are particularly well-suited for handling sequential input data such as natural language, as they can process the entire input all at once using the self-attention mechanism.

One of the most popular Transformer-based models for NLP is Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. 2019), which was trained on large language datasets such as the Wikipedia corpus and Common Crawl and can be fine-tuned for specific NLP tasks. Large language models (LLMs) such as the Generative Pre-trained Transformers (GPT) and LLaMA-Adapter are types of neural network architecture that are based on the Transformer model.

2.4. Review of Related Works

2.4.1. Sensing Modalities

The choice of sensor for AI-based technologies is determined based on the specific application for which it is intended to address. For instance, camera-based approaches are more beneficial when the sensor is optimally placed adjacent to DHH individuals, such as in delivering lectures in an educational setting. On the other hand, wearable sensors are found to be more productive when communicating socially, and camera sensors are counter intuitive in such cases.

2.4.1.1. Camera-based Approach

Al-Hammadi et al. (2020) carried out a study aimed at classifying Arabic sign language. The datasets used in this study were the Arabic sign language which was using three different cameras. The first dataset comprises of 40 signed words which were captured using RGB and Microsoft Kinect cameras, while the second dataset comprises of 23 gestures which were captured from three different participants. The third dataset comprises of 43 hand sign gestures. An accuracy of 98.12%, 100%, and 76.67% respectively was recorded on the single-dependent mode 3D-CNN model. However, an accuracy of 84.38%, 37.9% and 70% respectively was reported on the three datasets on the signer-independent mode model (Dere et al. 2023).

Aksoy, Salman, and Ekrem (2021) investigated the utilization of deep learning and image processing methods to detect the Turkish sign language. The researchers curated a dataset which comprises of 10233 images. A total of 29 letters in the Turkish sign language was collected. The researcher reported that CapsNet model achieved an excellent performance of 98.4%. The authors concluded that deep learning and image processing methods can be used to successfully detect Turkish sign language and this technology has the prospect of boost communication for people with DHH (Dere et al. 2023).

Mejía-Peréz et al. (2022) introduced models that could classify 30 distinct Mexican sign language collected with a depth camera using Long Short-Term Memories (LSTM) and Gated Recurrent Units (GRU). The study reported an accuracy exceeding 90% while illustrating the potency of LSTM and GRU-based models in sign language recognition through depth cameras (Dere et al. 2023).

Venugopalan and Reghunadhan (2021) applied DNN to recognize 13 words used by deaf farmers using a camera-based input sensor. To automate the classification of the words, the authors proposed a hybrid model of GoogleNet and BiLSTM sequence classifier. The study reported an average accuracy of 76.21% with a total training time of 7 hours. However, as with other camera-based studies, the basic limitation of this study is the ineffectiveness of the sensor placement. For optimal results, the camera must face the user to capture hand gesture movement, and the background can be an intricate issue that often degrades the effectiveness of camera-based computer vision applications (Dere et al. 2023).

2.4.1.2. Wearable-based Approach

A wearable sensor array which is fixed to the hand of a signer, along with a wireless printed circuit board and a speech synthesizer, was developed by (Zhou et al. 2020). Figure 1 shows the developed wearable sensor recognizing sign gestures and therefore interpreting text and speech. The shape and position of the signer's hand was detected using the sensor array, while the wireless printed circuit board was used to send the data to the speech synthesizer which then generates the corresponding spoken words. The developed system yielded an accuracy of 98.63% after been evaluated on 10 different participants. In addition, the system can translate signs into speech in less than 1 second. The authors concluded that the system has the prospect to bridge the communication gap between people living with DHH and normal hearing people.

However, it is important to note that the system is specifically designed for ASL (Dere et al. 2023).

Wen et al. (2021) introduced a novel triboelectric smart glove which was designed for sign language recognition and bidirectional communication in virtual reality (VR) settings. The glove was assembled with a deep learning model which has been trained to identify 50 vocabulary words and 20 sentences in ASL. During testing on a set of unseen sentences, an accuracy of up to 86.67% was achieved. The identified words are then converted to text and speech for application within a virtual environment. Figure 2 shows the developed sensor and the end-to-end framework for communication in VR (Dere et al. 2023).

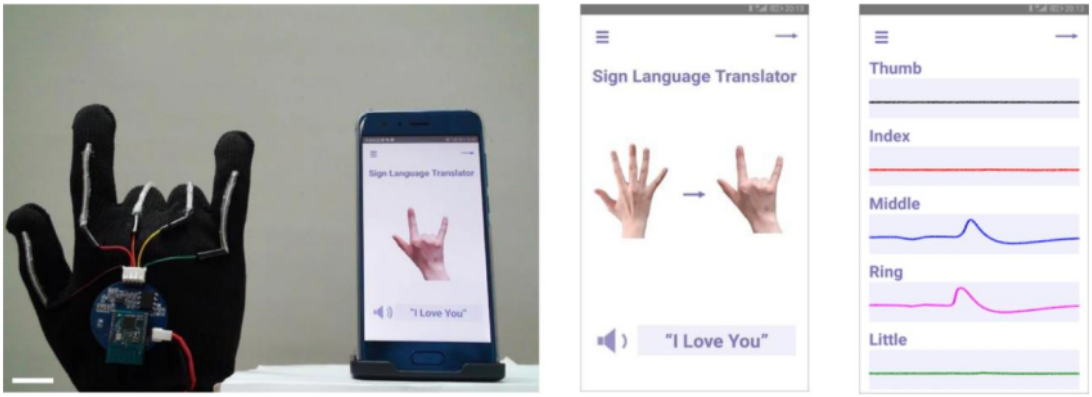


Figure 1. Illustration of the wearable sensor proposed by (Zhou et al. 2020)

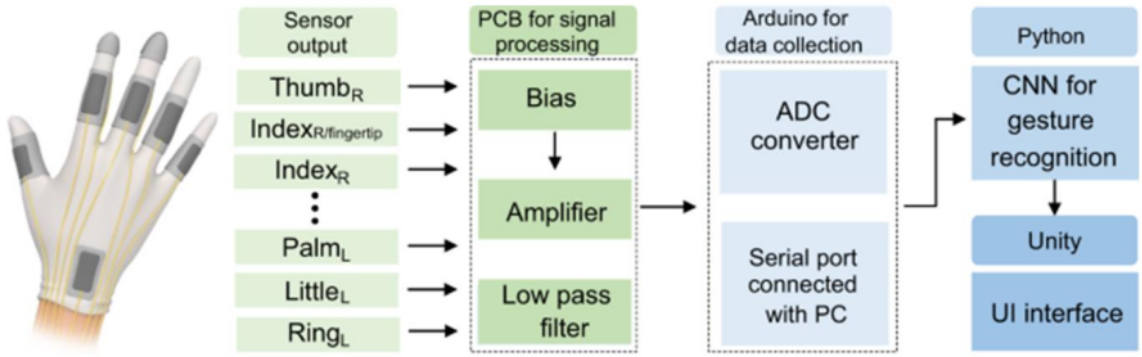


Figure 2. Wearable sensor including framework proposed by (Wen et al. 2021) for SLR.

Faisal et al. (2022) proposed an innovative approach of using cost effective datagloves for recognizing hand gestures as shown in Figure 3. The datagloves are built with five flex sensors and an IMU that has the ability of identifying 24 static and 16 dynamic gestures in ASL. A CNN architecture was implemented to carryout ASL recognition with an accuracy of 97.35%. This technology demonstrates significant potential for delivering cost-effective and accurate hand gesture recognition in sign language applications (Dere et al. 2023).

Sign language communication based on decoding patterns from IMU using HMM was proposed by (Liu et al. 2020). The HMM algorithm can decode signed gestures by individuals using the device shown in Figure 4. This is done by taking input from IMU sensors placed on a single finger and the wrist. The decoded gesture is then displayed on a mobile phone screen, enabling other individuals to read the intent of the signing user. The device design achieved an impressive average accuracy of 94.2% during validation on 10 users without hearing or speech impairment.

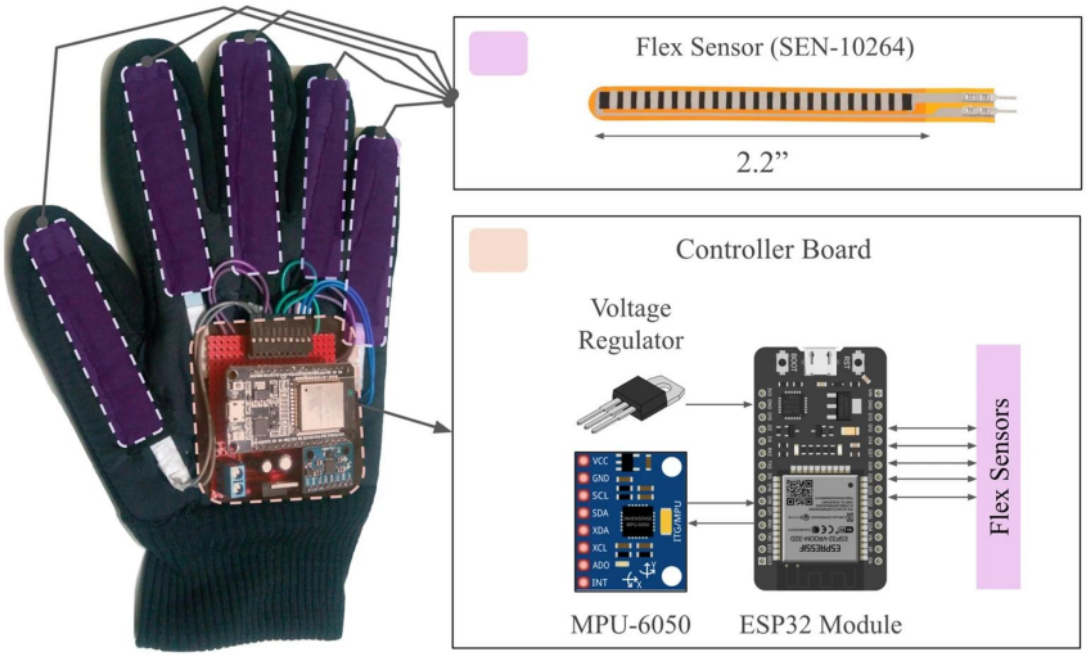


Figure 3. Dataglove developed by (Faisal et al. 2022) illustrating flex sensor.

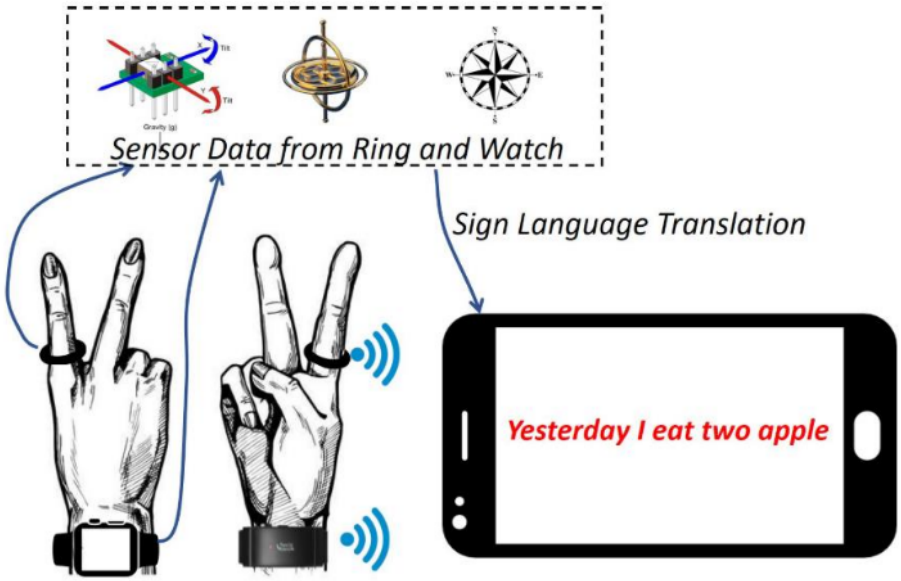


Figure 4. Design proposed by (Liu et al. 2020) for ASL communication.

2.4.2. Pattern Recognition Algorithm for SLR

Several conventional machine learning algorithms have been proposed for sign language communication (Haque, Das, and Kaspy 2019; Quinn and Olszewska 2019; Al Rashid Agha, Sefer, and Fattah 2018). However, conventional machine learning algorithms are constrained by domain-specific feature extraction. While deep learning algorithms have revolutionized pattern recognition, deployment for real-time inference on embedded devices is a major bottleneck.

XGBoost was proposed by Chen et al. (2021) was used for recognizing ASL from the features that were already extracted from the raw sEMG data. Three features were extracted in the time domain to classify four sign gestures irrespective of the pattern recognition algorithm been a lightweight, it was reported to have achieved a very low accuracy compared to the other similar deep learning approach. The study reported an overall average accuracy of 85% (Dere et al. 2022).

Venugopalan and Reghunadhan (2021) introduced a BiLSTM architecture that was designed for classifying 13 different agricultural-related words, it achieved an accuracy of 76.21%. Unlike other studies that use images, this study made use of video data from Indian sign language classification (Dere et al. 2022).

(Barbhuiya et al. 2021) implemented an improved AlexNet and VGG16 architecture for recognizing ASL. The experiment was designed to classify the 26 English letters with 10 digits. The authors recorded an outstanding classification accuracy of 99.82% with the AlexNet+SVM classifier and achieved an accuracy of 99.76% using the VGG16+SVM classifier (Dere et al. 2022).

2.4.3. Sign Language Recognition and Translation in INSL

Hassan et al. (2017) presented an ANN for SLR of 21 static, manual, and nonmanual HSL. This approach recorded an accuracy of 74.8% by manually extracting features, and 90.5% by using the particle swarm optimization algorithm respectively (Dere et al. 2023).

Kolawole et al. (2022) curated a dataset comprising of 137 words with 27 alphabet letters. The dataset was captured from 21 individuals making use of camera sensors. The study compared two CNN-models, in which the You Only Look Once (YOLO) model achieved the best performance and was deployed on an edge device for image-to-text and speech conversion (Dere et al. 2023).

Gueuwou et al. (2023) curated a dataset comprising of six different African sign languages including the NSL. The dataset consists of 152 hours of video recordings which was collected from videos obtained from the jehovah's witness (JW) sign language website. The study implemented a Transformer model, which achieved good performance on the dataset. The study contributes to the field of sign language translation by providing a new resource for low-resource African sign languages. The study also highlights the challenges and opportunities for future research on African sign language translation.

Olabanji and Ponnle (2021) developed a computer aided real-time interpretation system for indigenous sign language in Nigeria using CNN. The system consists of three modules: sign language recognition, sign language translation, and text-to-speech synthesis. The sign language recognition module uses a CNN model to classify sign language gestures from video frames. The sign language translation module uses a rule-based approach to convert sign language sentences

into English sentences. The text-to-speech synthesis module uses a pre-trained model to generate speech from text. The system achieved an accuracy of 96.67% for sign language recognition, for sign language translation. The system demonstrates the feasibility and potential of using CNN for sign language recognition and translation and provides a useful tool for enhancing the communication and education of deaf people in Nigeria.

2.4.4. Review Literature on SLRT

Arvanitis, Constantinopoulos, and Kosmopoulos (2019) used public dataset (ASLG-PC12) to translate American sign language glosses to English text. An encoder-decoder architecture with attention mechanism was used to translate the glosses to English text. The bilingual evaluation understudy (BLEU) score was adopted to evaluate the translation model. The study reported a BLEU-4 score of 0.65.

Cheng et al. (2023) proposed a novel framework, called CiCo, used for retrieving sign language. The task aims at finding the sign video or text that best matches the query in the other modality. The study developed sign language retrieval to serve as a cross-lingual retrieval task as well as a video-text retrieval task and introduces a cross-lingual contrastive learning method to model the fine-grained mappings between sign videos and texts. A domain-aware sign encoder was adopted which combines a domain-agnostic encoder pre-trained on large-scale sign videos and a domain-aware encoder fine-tuned on target datasets via pseudo-labeling, to extract discriminative and domain-aligned features of sign videos. The paper evaluates the proposed framework on three different sign language datasets, namely How2Sign (Duarte et al. 2021), PHOENIX-2014T, and CSL-Daily, and achieved a state-of-the art (SOTA) accuracy on How2Sign dataset (Dere et al. 2023).

Tarrés et al. (2023) presented baseline results for SLT on How2Sign, a large and broad dataset of instructional videos in ASL. The paper proposes a Transformer-based model that directly interprets videos into text, without using any means of intermediate medium. The paper also introduces reduced BLEU (rBLEU), a metric that is used for removing frequent and meaningless words from the reference and the prediction before computing the BLEU score, to enhance an excellent SLT performance. The paper reports a BLEU score of 8.03 and an rBLEU

score of 2.21 on the How2Sign test set and provides open code and models for reproducibility and further research (Dere et al. 2023).

2.5. Summary of Related Works

The related work examines wearable sensors for various applications, such as EMG, flex, and datagloves sensors. All the reviewed solutions achieve high accuracy (>90%), but two key limitations are identified: deploying DNN to embedded devices for inference and finding niche applications. This review focuses on sign language recognition for medical emergencies, an unexplored niche. The thesis proposes solutions to overcome these limitations and improve the performance of wearable sensors for this application which is low-cost.

Based on a review related works, it appears that all studies targeting the DHH in Nigeria have been limited using curated key word datasets which is inappropriate for Sign-to-Text applications in education. Furthermore, previous studies validated models using application-specific metrics, making it difficult to compare to benchmark datasets. However, this thesis proposes a novel approach to mitigate these constraints and enhance higher accuracy of Sign-to-Text models for educational settings. The approach is unique and has not been previously explored to the best of my knowledge.

CHAPTER THREE

METHODOLOGY

3.1. Preamble

To address the issue of communication gaps between individuals who are DHH and those who are not, we propose two methods in this study.

The first method is referred to as SmartCall. The SmartCall is a wearable device that could be worn by DHH individuals to communicate during period of medical emergencies. As investigated in "Review of Related Works", AI-based technologies have been developed for niche applications such as farming (Venugopalan and Reghunadhan 2021), but there is currently no research explicitly focused on enabling individuals who are DHH to communicate during medical emergencies. To address this gap, we introduce SmartCall, a wearable device designed to facilitate communication for DHH individuals in emergency medical situations.

Conventional machine learning algorithms have been established to be insufficient in achieving optimal performance without domain-specific feature extraction. Although DNN can alleviate this limitation, their deployment to embedded devices is challenging, with most literature relying on Internet of Things (IoT) for communication between devices, leading to latency and data privacy concerns. To address these limitations, we deployed DNN on an embedded device to enable on-device inference without the need for IoT connectivity.

The second method, known as SmartCom. The SmartCom method is an AI solution that utilizes camera sensors to integrate individuals who are DHH into diverse aspects of the Nigerian economy, including education and media. Our approach to this challenge involves generating English text form ASL video using a pre-trained CNN model coupled with a Transformer model. The resulting English text is then passed to a LRL translation model to interpret the English text

into Hausa, Ibo, and Yoruba languages. Our approach is novel because previous studies have focused on curating INSL datasets to train models for SLR. However, this approach may not be suitable as there is yet to be an officially recognized NSL. Furthermore, the previous approach overlooks the need for communication with normal hearing people in their indigenous languages, which is where SmartCom bridge communication gap.

ASL was chosen as the appropriate sign language for our study because NSL extensively borrows vocabulary from ASL. Additionally, ASL is the most widely used sign language worldwide among DHH individuals. Hausa, Ibo, and Yoruba were chosen because they are the most spoken languages in Nigeria (Sasu 2022). The significance of SmartCom is that individuals with DHH impairments can be integrated into traditional school setups to teach normal hearing individuals, as well as provide employment opportunities in media houses where they can serve as primary presenters for media content.

3.2. Problem Formulation for SmartCall

We adopted a supervised learning approach, which involved two steps, as depicted in Figure 5. The problem SmartCom is meant to solve is a distinct class recognition, thus the formulation of the problem as supervised learning. Supervised learning is a commonly used approach for developing machine learning models for SLR tasks. In supervised learning, the model is trained on a labeled dataset, with the input data (IMU) associated with pre-defined output labels (emergency words).

3.3. Proposed Solution by SmartCall

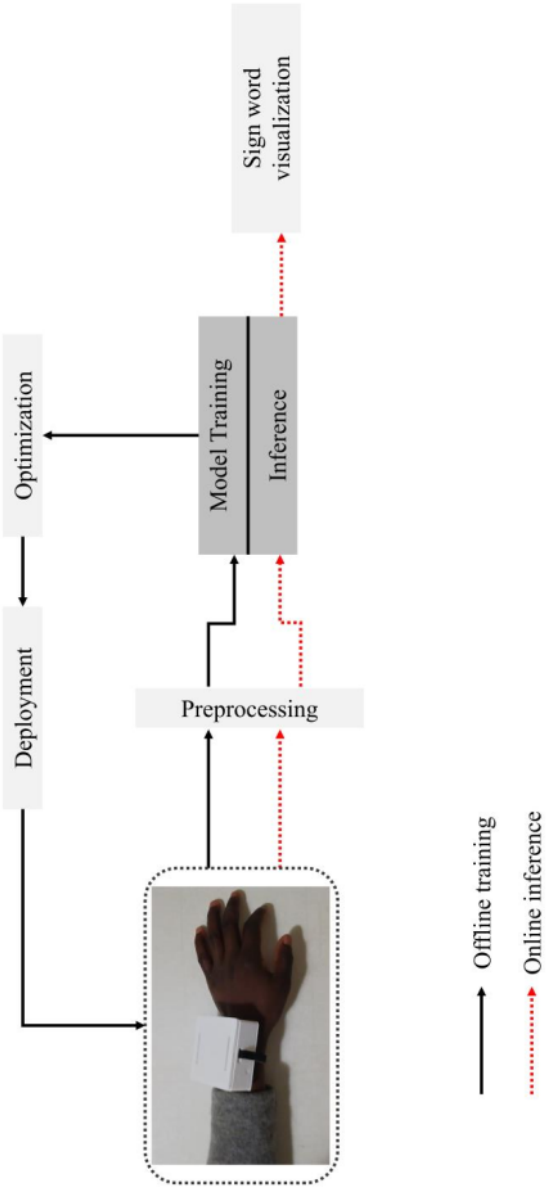


Figure 5. Overall workflow of SmartCall.

3.4. Tools used in the Implementation of SmartCall

The implementation of SmartCall is divided into steps: The offline and the on-device steps. In the offline step: Data collection, labeling, preprocessing training, and offline accuracy were performed using Edge Impulse on a PC. In the on-device step, the trained model was deployed to an embedded device, and the on-device performance was evaluated during inference.

Edge Impulse is an edge machine learning platform that enables developers to build, train, and deploy machine learning models on edge devices (Hymel et al. 2022). The platform provides a complete workflow for machine learning development, from data collection and preparation to model training and deployment. Edge Impulse supports a wide range of edge devices, including Arduino Nano BLE. Edge Impulse uses TensorFlow and TensorFlow Lite for training, optimization, and deployment of the model to the edge. TensorFlow is a Python-based framework for deep learning workload (Leon et al. 2020).

3.5. Approach and Technique for the SmartCall

SmartCall is a supervised learning approach used to recognize words from an embedded device. The supervised learning paradigm includes several popular models such as CNN, RNN, ANN, and Restricted Boltzmann Machines. Though RNN is often preferred for time series data, such as IMU data, it is computationally expensive and not suitable for deployment on an embedded device with limited resources due to its lack of parallelization, which causes training and deployment to be a bottleneck. On the other hand, RBM and ANN are suboptimal because of the contrastive learning mechanisms used. 1D-CNN, which is capable of learning spatiotemporal information and sequences in temporal data, is a better choice. Additionally, 1D-CNN is parallelizable and can be easily optimized for deployment on edge devices, making it a more practical option.

Figure 6 shows the 1D-CNN architecture proposed for the emergency word recognition. The convolutional layers were designed to automatically learn temporal features relevant to sign word classification. The given equation (Equation 1) represents a convolution operation that is typically used in deep learning models. It takes an input signal (I), which can be the raw data from a previous layer or an IMU and applies a kernel (F) with shape (m, n) to convolve the signal and extract relevant features. In the 1-D case, the kernel is typically a vector with the length (m) set to 1. The output of the convolution operation is denoted by S, and its value at a given coordinate (i, j) is determined by computing the sum of the element-wise products between the input signal (I) and the kernel (F) after sliding the kernel across the input signal. This process captures how the input signal relates to the kernel, and the output of this convolved signal can be used as input to subsequent layers in the network to learn more complex representations of the input data.

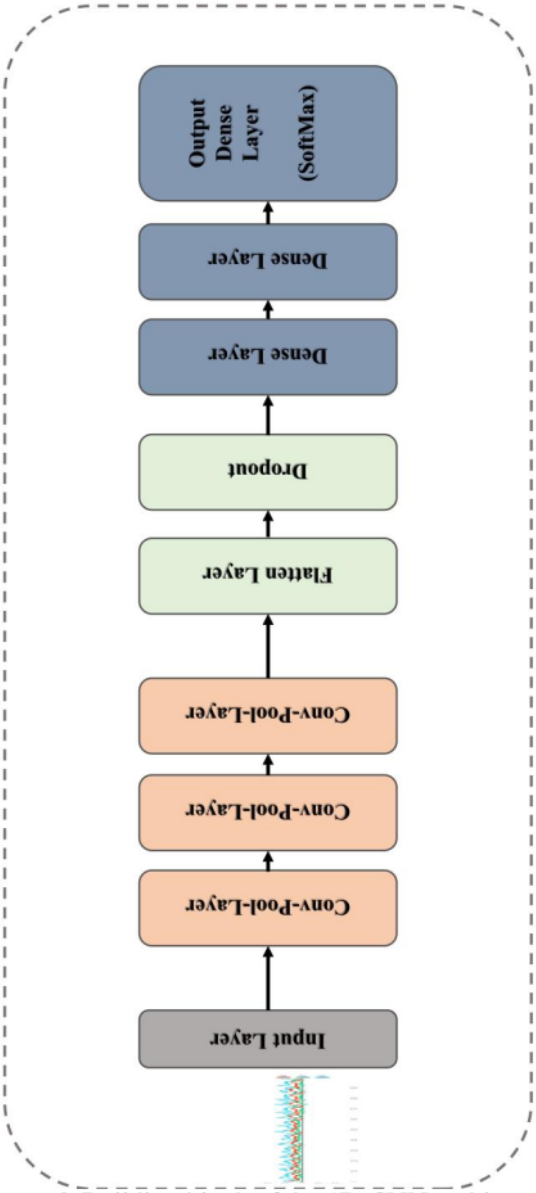


Figure 6. Building block of the 1D-CNN architecture.

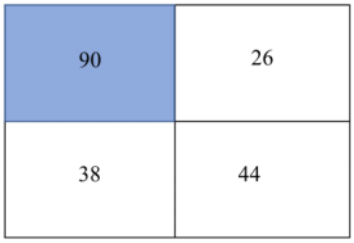


Figure 7. MaxPooling operation. The shaded area indicates the maximum value.

The MaxPooling operation is used with at the end of every convolution layer as show in Figure 7. The main function is to reduce the spatial dimensions of the input feature map while retaining the most relevant information. This is accomplished by partitioning the input feature map into non-overlapping regions and computing the maximum value within each region as shown in Figure 7. The resulting output feature map has reduced spatial dimensions due to the pooling but preserves the most important features, as the maximum value represents the strongest activation in that region. Maxpooling also helps the network become more invariant to small changes in the input data, leading to improved generalization and performance.

In Figure 6, three convolution-maxpooling layers are shown. The output feature map from the third maxpooling layer is transformed into a 1-D vector form, as the downstream fully connected layer known as the Dense layer can only process 1-D vector data. This flattening operation is necessary for further processing of the feature maps.

The Dropout layer is a type of regularization technique used in deep learning to reduce overfitting. Dropout works by randomly dropping out some percentage of the input units in each layer during each training iteration. This forces the network to learn more robust features that are not reliant on any single input unit, as units that are dropped out do not contribute to the forward

pass or backpropagation. As a result, the network becomes less likely to overfit to the training data and

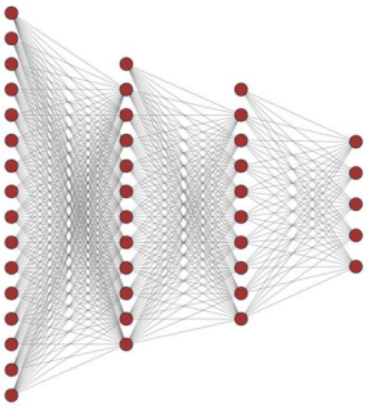


Figure 8. Visualization of the fully connected layer also known as Dense layer.

has improved generalization performance on new data. A dropout probability value of 0.5 was used in training the SmartCall model. During a forward pass operation, the Dropout value of 0.5 implies that approximately 50% of the neurons will be active, and the other 50% will not be active. The dropout layer is utilized solely during the training phase of the SmartCall model, after which it is removed for the inference model.

The dense layer (Figure 8) processes the features map from the previous layer with a set of learned weights, followed by an activation function. The model's dense layer learns non-linearity, while the final layer performs the classification task using SoftMax activation function (Equation 2). Each one of the training and optimization were conducted using the Edge Impulse. The SoftMax function weighs the output between 0 and 1, taking in an input vector Z with elements j and K classes. Except for the final dense layer, every other layer in the model made

use of Rectified Linear Unit (ReLU) activation function as described in Equation 3. The ReLU function hastens the convergence speed and helps solve the vanishing gradient problem (Dere et al. 2022).

The SmartCall model was trained with a cross-entropy loss function (CE) (Equation 4). The cross-entropy loss function is commonly used in machine learning for classification tasks. Cross-entropy loss function measures the distance between two probability distributions, typically the predicted distribution output by the model and the true distribution (i.e., the ground truth). K in Equation 2 and Equation 4 represent the classes (word) in the dataset.

$$S(i,j) = (I * K) = \sum_m \sum_n I(i+m,\ j+n)K(m,n) \quad (1)$$

$$\text{SoftMax}(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad (2)$$

$$ReLU(I) = max(0,\ I) \quad (3)$$

$$CE = -\sum_{n=1}^{k} \overline{y_n}\ log\ y_n + (1 - \bar{y}_n) log(1 - y_n) \quad (4)$$

3.6. SmartCall Research Design

We adopted a supervised learning approach, which involved two steps, as depicted in Figure 5. In the first step, data collection, labeling, and preprocessing were performed, and training and offline accuracy performance were evaluated. The second step involved deploying the already trained model on an embedded device, and the evaluation was carried out on the on-device performance during the inference (Dere et al. 2022).

The study was undertaken with the objective of authenticating the design of an affordable embedded sign language communicator through a series of three experiments. The initial experiment was executed to ascertain the offline accuracy of the trained model using a personal computer lacking a GPU. Subsequently, the second experiment was conducted to assess the on-device accuracy of the model upon deployment to an Arduino Nano 33 BLE (Arduino Official Store 2020). Finally, the third experiment delved into investigating the transferability of the model across individuals (Dere et al. 2022).

3.7. Description of Validation Techniques for SmartCall

3.7.1. Data Collection

Data was gathered from the Arduino Nano 33 BLE onboard IMU using the Edge Impulse software. The research included two participants, both 29 years old. User one had no speech impairment, while user two had a DHH impairment. User one received training in proper Signing from a volunteer with speech impairment and then practiced by watching on-device ASL tutorial videos. All participants provided informed consent after being briefed on the experimental protocol. The embedded device was fastened to the right-hand wrist, and data was collected at a frequency of 100 Hz while signing the emergency words in the dataset for 60 seconds each (Dere et al. 2022).

To evaluate the cost-effective embedded Sign language communicator, five medical emergency words such as help, hospital, ambulance, doctor, and call, were chosen after comprehensive online research and consulting medical experts. Before deploying, the data acquisition required signing each of the words for 10 seconds to test the trained model (Dere et al. 2022).

3.7.2. Preprocessing

Data augmentation refers to the process whereby the number of training data samples are being increased to achieve generalization and to avert overfitting. The raw time-series IMU data, depicted in Figure 9(a), was segregated into smaller data points using a sliding window of 100ms with a 50ms overlap, resulting to a total of 5970 data points (Dere et al. 2022). The segmented IMU features were subsequently visualized, as illustrated in Figure 9(b), to analyze the pattern clusters of the signed words. It was noted that the signed words formed well-clustered patterns, which is crucial for pattern recognition.

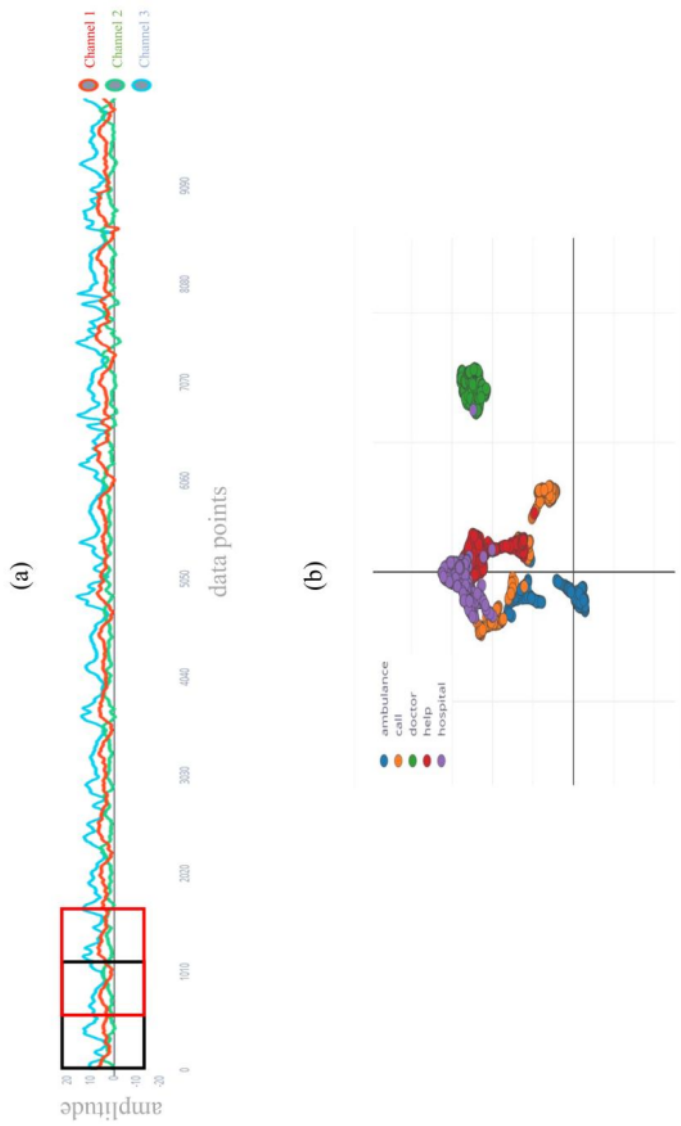


Figure 9. Visualization of the: (a) Raw IMU data temporally. (b). Data in the spatial domain

3.7.3. Offline Training

SmartCall uses a supervised learning approach to recognize the words signed. In supervised learning, machine learning algorithms translate the extracted features from the dataset to their corresponding labels using a process called backpropagation. A 1D-CNN model (Figure 6) was trained on the dataset and optimized for deployment to an Arduino-embedded device using Edge Impulse.

3.7.4. On-device Deployment

The trained model was optimized with Edge Impulse by quantizing from float-32 to fixed-point Int-8. Quantization of the model compresses the model weight matrices and reduces the precision of its computations, to optimize the network for the resource-constrained on the Arduino Nano BLE (Arduino Official Store 2020).

3.8. Description of Performance Evaluation of SmartCall

3.8.1. Offline Accuracy Evaluation

The acquired dataset was split into a training dataset and a test dataset, with 80% of the data allocated for training and 20% for test. This hold-out method is commonly used to train DNN. The training dataset was used to train the model utilizing a three-fold cross-validation approach. This approach helps to prevent the model from becoming biased by analyzing the evaluation metrics on different folds of the data. Finally, the test dataset was employed to assess the offline accuracy of the trained model by using it as an unseen dataset for the trained model to make predictions and comparing those predictions to the actual values. The use of these different datasets and approaches ensures that the trained model performs well on unseen data and is not overfitting the training data.

The trained model accuracy was accessed offline using Equation 5 and Equation 6, with the test dataset from user one being used in experiment one. Where TP is the True Positive, FP is the False Positive, FN is the False Negative, and TN is the True Negative. TP refers to an accurate prediction of a positive gesture, while TN refers to an accurate prediction of a negative gesture. FP occurs when a negative gesture is predicted as a positive one, and FN occurs when a positive gesture is predicted as a negative one. The F1-score was computed using Precision (Equation 7) and Recall (Equation 8).

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100 \quad (5)$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (6)$$

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

3.8.2. On-device Accuracy Evaluation

To evaluate the model during real-time inference, Equation 9 was employed, and five repetitions of all the words in the dataset were signed by the user. Experiment two involved user one carrying out the real-time evaluations, while in experiment three, the model was trained with data from user one, but evaluated in real-time with user two.

$$On - device\ Accuracy = \frac{\text{Correct Classification}}{\text{Total Number of Samples}} \times 100 \quad (9)$$

3.9. SmartCall System Architecture

The fine-tuned model was implemented on an Arduino Nano BLE that is assembled with a three-channel accelerometer, gyroscope, and magnetometer. The magnetometer is used to measure wrist positioning and orientation, while the accelerometer tracks hand acceleration during the signing of words. The gyroscope records rotational data of the wrist. Figure 9(a) displays the IMU data from these three different channels. To assess the deployed model, the Arduino device was connected to a serial monitor to supervise the classification output (Dere et al. 2022).

A 3D printed square box was printed to house the Arduino device as shown in Figure 5. A printer was used in creating a square box to house an Arduino device, which was fitted with a strap to secure it to the user's wrist. One significant advantage of using 3D printing for this application is its relatively low cost compared to traditional manufacturing methods. 3D printing allows for the creation of complex geometries and customized designs with high levels of accuracy at a relatively low cost. This makes it an ideal choice for producing low-volume and prototype products, as well as for creating customized solutions such as the strap designed for the Arduino device. In addition, 3D printing enables quicker design iterations, allowing for quicker optimization of the product. The use of 3D printing technology for this application demonstrates its adaptability to a wide range of industries and applications, from healthcare to consumer electronics.

3.10. Problem Formulation for SmartCom

SmartCom is dedicated to developing a comprehensive framework for the automated interpretation of ASL to three major spoken languages in Nigeria, namely Hausa, Ibo, and Yoruba. The framework, as depicted in Figure 10, comprises two models utilized in the study: a sign gesture recognition model that translates video input into English text, and the No Language Left Behind (NLLB) model (NLLB Team et al. 2022), which has been fine-tuned to convert American English text into the textual representation of the Hausa, Ibo, and Yoruba languages.

3.11. Proposed Solution by SmartCom

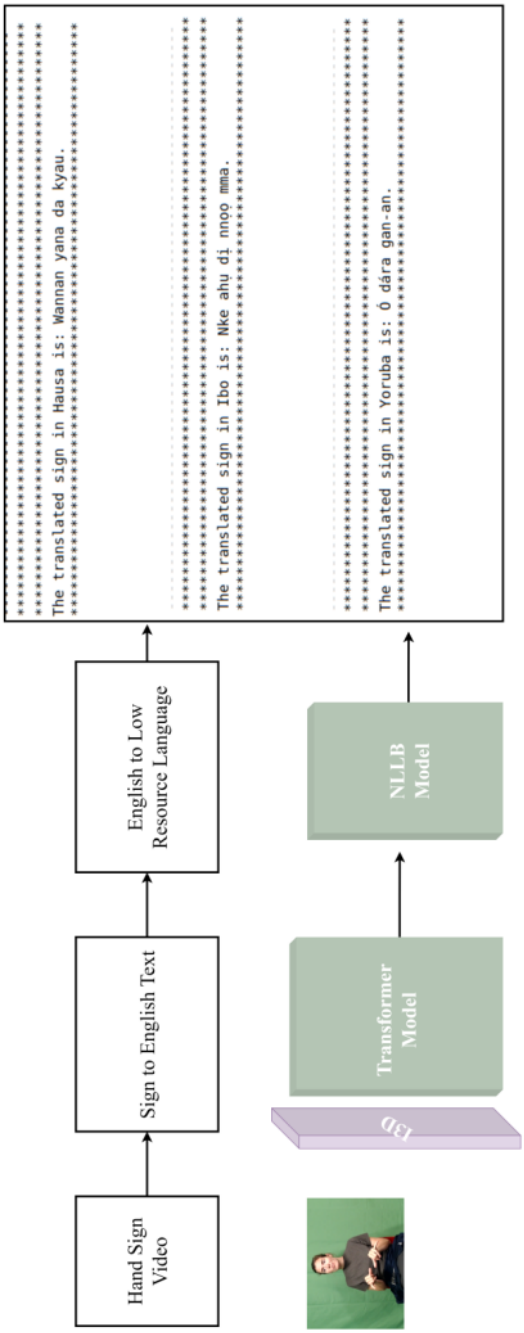


Figure 10. End-to-end framework proposed for SmartCom.

3.12. Tools used in the Implementation of SmartCom

The SmartCom training, inference, and evaluation were carried out on an 8-core Intel PC with a single GPU (RTX-3090) operating on a Linux-based system. The utilization of a GPU significantly enhances the training process for machine learning models, including deep learning models, by enabling expedited training times. Additionally, the employment of multiple cores and a Linux operating system contributes to the efficiency of the training process through the facilitation of parallel computations and optimization.

In building and training the machine learning model, Python was the preferred programming language, with the open-source deep learning framework PyTorch being comprehensively utilized all through the implementation process. PyTorch is a widely adopted machine learning framework that offers robust support for the creation, training, and deployment of deep learning models. Furthermore, OpenCV, a library primarily focused on computer vision, was extensively employed in the preprocessing phase, providing a wide array of tools and utilities for processing images and videos, as well as detecting and tracking objects. In the preprocessing of the text dataset, Torchtext, a PyTorch library, was utilized, offering a variety of preprocessing utilities and popular datasets for NLP. Through the utilization of these open-source tools, developers can construct and train advanced machine learning models with relative ease and efficiency.

3.13. Approach and Technique for the SmartCom

3.13.1. Sign-to-Text Model

The utilization of a Transformer model facilitated the iterative acquisition of information from the transcript to develop words in America English. The implementation of the Transformer's architecture, as illustrated in Figure 11 featured an encoder-decoder structure. The I3D model was employed to produce a chain of outputs, which serves as input to the encoder block. Within the encoder-decoder block consist of a multi-head attention (MHA), layer normalization (LN), and feed-forward (FF) layers were incorporated to undertake the input and yield significant representations (Dere et al. 2023).

MHA stands as a pivotal element of the Transformer architecture, facilitates the model to understand the connection between the input sequence. This is accomplished through the projection of the input sequence into concealed representations, followed by the computation of attention weights (Heads) between each pair of the concealed representations. Subsequently, these attention weights are utilized to integrate the concealed representations into a singular output representation. One of the notable advantages of this approach lies in the model's ability to discern relationships among disparate segments of the sequence, irrespective of their spatial arrangement. The model can also discern multiple diverse relationships among the same segments of the sequence, as the MHA layer computes attention weights for each pair of concealed representations and integrates them in various manners (Dere et al. 2023).

A single head (self-attention) of the MHA is computed using Equation 10, the query (Q), key (K), and value (V) are projections of the input sequence, and the projection dimension is denoted by d_k . The attention weights are computed using an attention function that takes in Q, K, and V, and outputs a weighted sum over the values, with the weights determined by the

attention scores between Q and K. The self-attention mechanism allows the model to selectively focus on relevant parts of the input sequence to generate context-aware embeddings. Overall, the MHA layer with self-attention is a powerful computational tool that has significantly improved the performance of natural language processing tasks (Dere et al. 2023).

The LN is used together with the encoder-decoder block of the Transformer to ensure the stability during the training process. Prevention of the gradient from becoming too large or too small is done by normalizing the output of each layer. Such normalization is crucial, as gradients of this nature can impede the model's learning process. The process of LN entails the computation of the mean and standard deviation of a layer's output for the purpose of normalization. The resulting normalized output, represented as LayerNorm(X) is evaluated using Equation 10, where X denotes the input of the layer, μ represents its mean, σ signifies its standard deviation, ϵ is a minute constant utilized to prevent division by zero, and γ and β denote learnable scaling and shifting parameters, respectively. These parameters are applied element-wise through the Hadamard product ($\odot$) (Dere et al. 2023).

The FF layer serves to effect a linear transformation on its input, subsequently applying a ReLU non-linear activation function (Equation 3). Within the Transformer architecture's encoder-decoder block, the FF (Dense) layer is employed to convert the output of the MHA layer into a novel representation conducive to acquiring intricate non-linear relationships (Dere et al. 2023).

$$Attention(Q,K,V) = SoftMax(\frac{QK^T}{\sqrt{d_k}})V \quad (10)$$

$$LayerNorm(X) = \gamma \odot \frac{X-\mu}{\sigma+\epsilon} + \beta \quad (11)$$

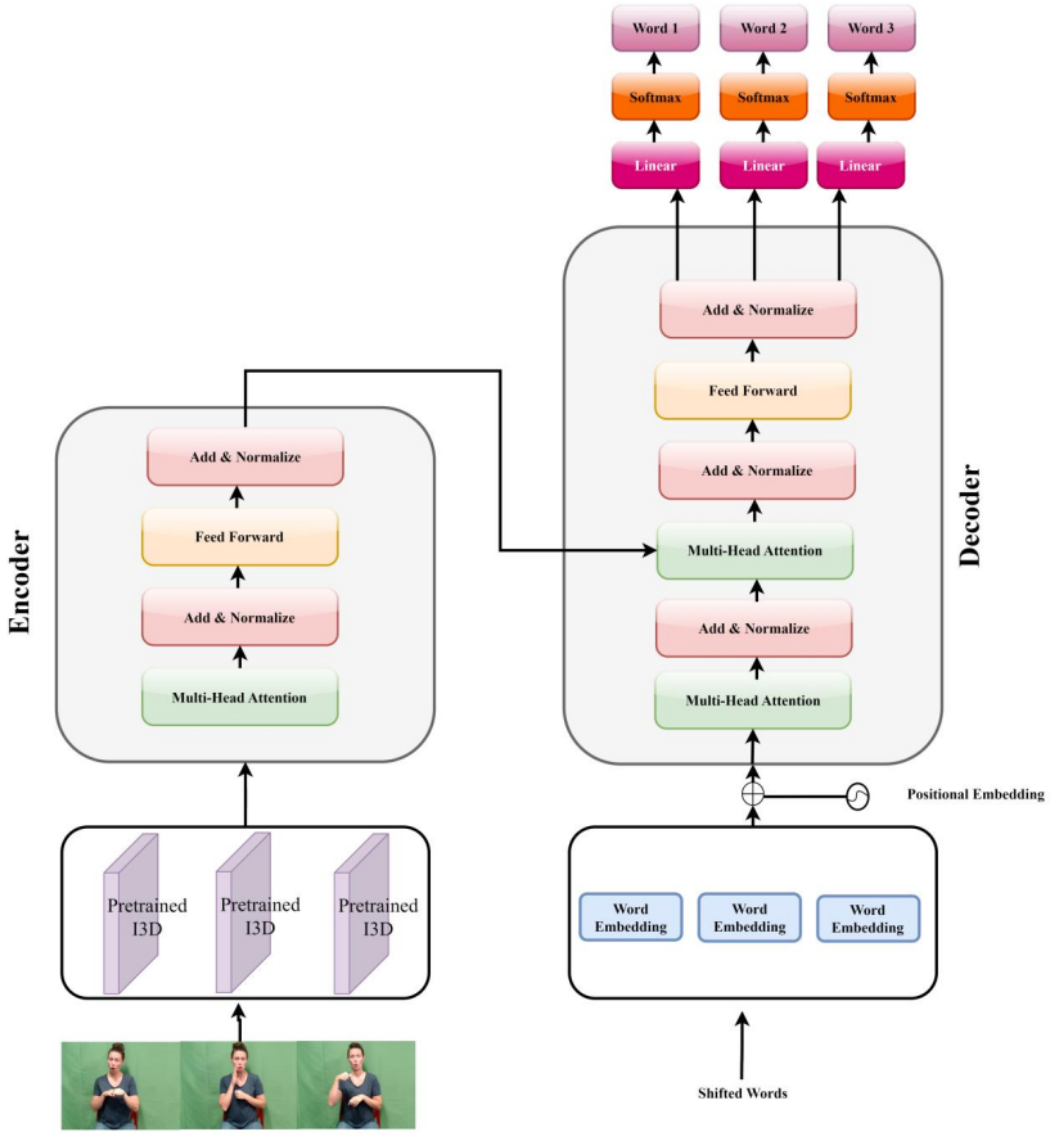


Figure 11. The Sign-to-Text model is used to generate text from sign video.

3.13.2. Text-to-LRL Model

The NLLB open-source model (NLLB Team et al. 2022) demonstrates proficiency in translating to 200 LRL. Implementing a Transformer architecture trained on the FLORES-200 dataset from Meta AI, the NLLB model has the ability of breaking down the language barriers and enhance global access to information. Its utility extends to the translation of diverse content, encompassing news articles and educational materials, into languages that are not widely supported by other machine translation systems. This capability contributes to ensuring universal availability of critical information, irrespective of individuals' linguistic backgrounds. As far as my knowledge extends, the NLLB model has not been assessed for its performance in ASL-to-LRL translation (Dere et al. 2023).

.

3.14. SmartCom Research Design

The validation of the SmartCom framework entails both quantitative and qualitative analyses. The quantitative assessment encompasses the evaluation of the test set using the BLEU score across various hyperparameters outlined in Table 1. Subsequently, the model delivering optimal performance, as per the outcomes, is selected for deployment in the qualitative analysis (Dere et al. 2023).

The qualitative evaluation of the SmartCom framework involves the enlistment of three native speakers (participants) for each LRL. Five videos are randomly chosen from the test dataset, with each video undergoing processing by the Sign-to-Text model and subsequently the NLLB model. The resultant output of the SmartCom framework is shown within a Jupyter notebook, and the evaluation of translation quality is conducted via a questionnaire (refer to Appendix I). The qualitative analysis comprises two segments: Firstly, native speakers are presented with translated text in low-resource languages and are then prompted to assess comprehension using the provided questionnaire. Secondly, participants are presented with the original English text, the English text generated by the Sign-to-Text model, and the Text-to-LRL text. Subsequently, they are requested to evaluate the translation performance of both the Sign-to-Text and Text-to-LRL models (Dere et al. 2023).

3.15. Description of Validation Technique for SmartCom

3.15.1. Dataset Description

The SmartCom framework leverages the How2Sign public dataset (Duarte et al. 2021) for training the video-to-text model. This dataset encompasses over 80 hours of sign language videos captured from diverse perspectives, accompanied by speech, English transcripts, and depth information. A subset of 3 hours was specifically recorded in the Panoptic studio to facilitate 3D pose estimation. The dataset was sourced from 11 signers, including 6 individuals who identify as DHH. Notably, our study utilized the frontal view camera dataset, featuring a resolution of 1280x720 and a frame rate of 30 fps, captured using a depth and high-definition camera (Dere et al. 2023).

Furthermore, the NLLB model underwent training on the FLORES-200 dataset, which stands as a benchmark dataset for facilitating machine translation between English and low-resource languages. Curated by META AI, this dataset is designed to bolster the development of human-centric machine translation models capable of catering to over 200 languages.

3.15.2. Video Preprocessing

The video data went through the preprocessing stage, followed by alignment with corresponding transcripts through the utilization of timestamps. We executed data preprocessing by resizing the original video data from 1280x720 to 224x224 to enhance the efficiency and generalization of the deep learning model (Dere et al. 2023). Subsequently, we employed a pre-trained inflated 3-dimensional (I3D) model to extract features from the resized video files and was utilized to improve accuracy. The pre-trained I3D model (Carreira and Zisserman 2017) harnesses a CNN architecture to extract features from the second-to-the-last layer, subsequently summing to produce two tensors encompassing 1024-d features for RGB and flow streams.

Notably, the I3D model was trained using the Kinetics 400 dataset put forth by (Smaira et al. 2020).

This approach was implemented to attain notable accuracy, with the resulting output saved in a NumPy array format characterized by a shape of (video length x 1024). The delineation of the training, validation, and test set split employed in all experiments within this study is depicted in Figure 12. Furthermore, the video preprocessing procedures were executed utilizing OpenCV.

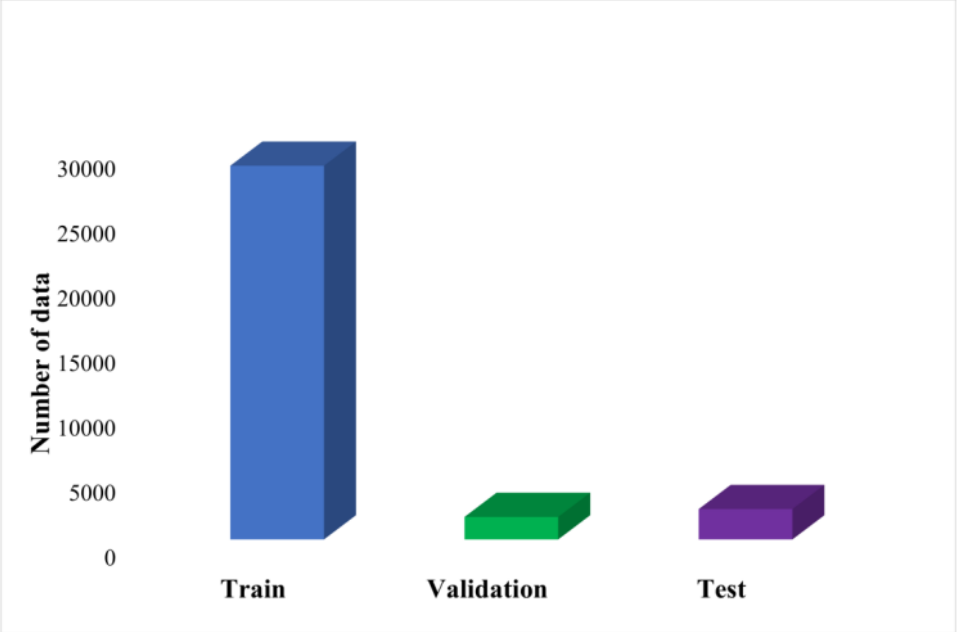


Figure 12. How2Sign dataset split after video preprocessing.

3.15.3. Text Preprocessing

The preprocessing of text holds paramount importance in achieving superior accuracy for a translation model. Within the context of the How2Sign dataset, the aligned sentences were initially converted to lowercase before being employed as glosses for the translation model.

Glosses function as an intermediary representation bridging sign images and output text. Notably, (Othman and Jemni 2019) introduced a statistical model for generating glosses from the transcript, given the absence of glosses within the How2Sign dataset at present. This method paves the way for enhancing the accuracy of forthcoming sentence segmentation and translation models for sign language.

3.16. Description of Performance Evaluation

3.16.1. Training and Evaluation

Figure 12 illustrates the distribution ratio of the dataset split, providing insight into the allocation of data for training, validation, and testing. Meanwhile, Table 1 comprehensively presents all the hyperparameters utilized across the experiments conducted in this study. The evaluation of the sign-to-text model involved the utilization of the BLEU score (Papineni et al. 2002) with n-grams 1, 2, 3, and 4. The model was trained on an 8-core Intel PC with a single GPU, operating on a Linux-based system. Throughout the training process, the early-stopping approach, with a patience of 8 epochs, was consistently implemented across all experiments. Notably, the models investigated in this study share certain common hyperparameters, as detailed in Table 2.

Table 1. Hyperparameters were used to train the different Sign-to-Text models.

	Hyperparameter	Value
Model 1	Scheduling	Cosine annealing
	Embedding dimension (Encoder, Decoder)	(512, 512)
	Number of layers (Encoder, Decoder)	(3, 3)
	Number of heads	(8, 8)
Model 2	Scheduling	Noam
	Embedding dimension (Encoder, Decoder)	(256, 256)
	Number of layers (Encoder, Decoder)	(3, 5)
	Number of heads	(8, 8)
Model 3	Scheduling	Plateau
	Embedding dimension (Encoder, Decoder)	(512, 512)
	Number of layers (Encoder, Decoder)	(5, 3)
	Number of heads	(16, 8)
Model 4	Scheduling	Plateau
	Embedding dimension (Encoder, Decoder)	(512, 512)
	Number of layers (Encoder, Decoder)	(3, 3)
	Number of heads	(8, 8)

Table 2. Hyperparameter similar to all models

Hyperparameter	Value
Initialization	Xavier
Dropout	0.1
Training epochs	500
Early stopping patience epochs	8
Optimizer	Adam

CHAPTER FOUR

RESULTS

4.1. Preamble

The results are presented in two sections: Section one reports the result for the SmartCall design proposed in the first study, while Section two reports the result for the SmartCom framework proposed in the second study. Additionally, the results are presented in accordance with the research questions.

4.2. SmartCall System Evaluation

In this study, the results are presented both offline and on-device. The offline validation accuracy serves as a measure of how well the model generalizes to unseen data and helps to detect any issues with underfitting or overfitting to the training dataset. This offline evaluation provides important insights into the performance of the model before it is deployed for use. The validation dataset is used for tuning the hyperparameters that ultimately influence the performance of the model. The optimal model is obtained by using the best set of hyperparameters, determined through a process of trial and error, in combination with the validation dataset. By presenting the results in this manner, we can demonstrate the effectiveness and generalization capability of our model and provide insight into how it can be further improved in the future. Overall, emphasis is placed on robust and thorough evaluation to ensure that the model is performing at a high accuracy and is ready for deployment in real-world scenarios.

Deep learning models have become increasingly popular and powerful in recent years, providing accurate predictions, and helping to enable many innovative applications. However, the models may perform differently when they are deployed offline or on embedded devices (on-device), where resources are limited. Therefore, it is crucial to evaluate the accuracy of deep learning models offline and on embedded devices for real-time inference. Offline evaluation allows us to identify potential issues with overfitting or underfitting the training dataset, while on-device evaluation helps us understand how well the model will perform in real-world scenarios. Both offline and on-device evaluation can be achieved by using Equation 5 (Accuracy), Equation 9 (On-device accuracy), Equation 6 (F1-score), Equation 7 (precision), and

Equation 8 (Recall). By evaluating deep learning models in these ways, we can ensure that they are robust, reliable, and perform at a high level in practical applications.

Confusion matrices are an effective tool for evaluating the accuracy of a classification model as they provide a more detailed breakdown of the model's performance than a simple accuracy score. Specifically, a confusion matrix tallies the number of true positives, true negatives, false positives, and false negatives for each class or category. By examining these values, we can calculate a range of performance metrics such as precision, recall, and F1 score, which provide a more nuanced and informative picture of the model's accuracy. Additionally, by evaluating a model's accuracy using a confusion matrix, we can identify which categories or classes are being misclassified most frequently and adjust our model accordingly. This can help us optimize our model's accuracy for specific applications and ensure that it is performing effectively in real-world scenarios.

We measure the classification latency of the deployed model on-device. Latency refers to the time it takes for a model to process input and produce an output. Latency is an important metric to measure during inference as it can have a significant impact on the performance of a model in real-world applications. The latency is measured by profiling the classification time in between consecutive inference. The metric of measuring latency is in milli-seconds (ms).

4.3. SmartCall Result Presentation

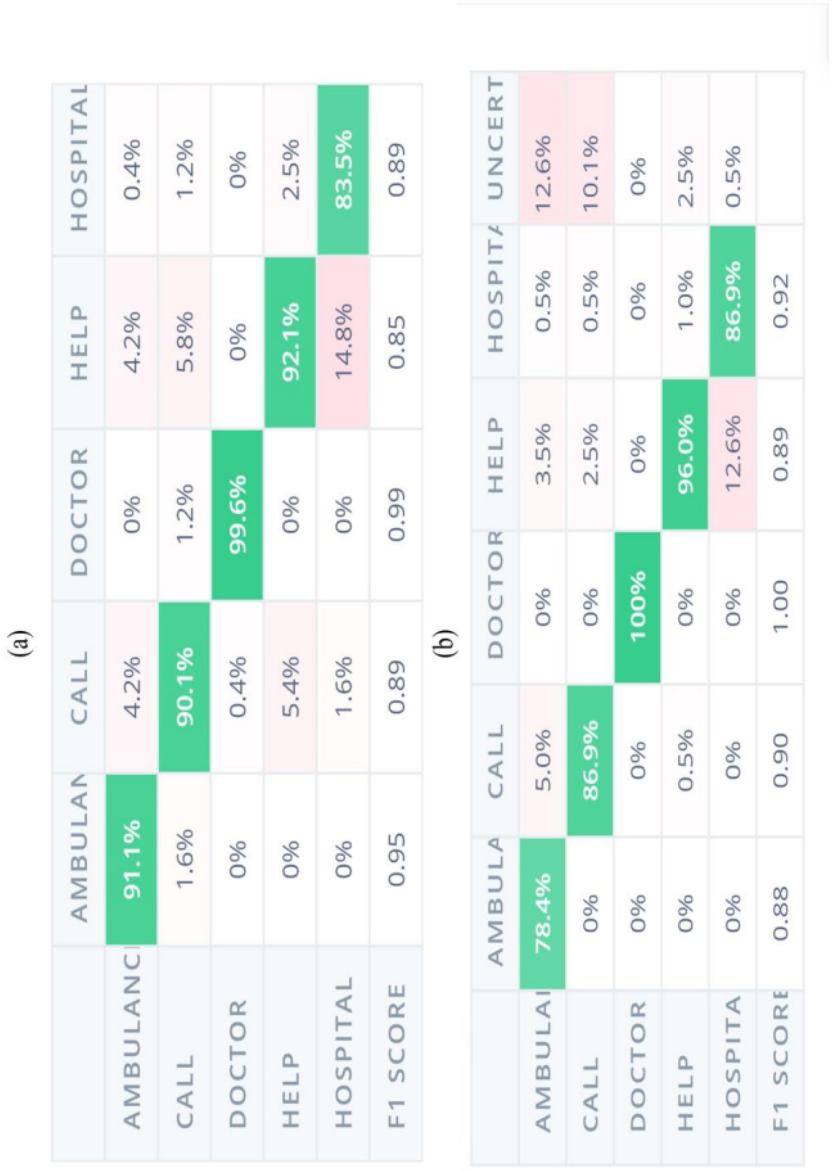


Figure 13. SmartCall confusion matrices. (a) Offline validation results. (b) Offline test results.

Figure 14. Comparison of the quantized and unquantized model on the test dataset.

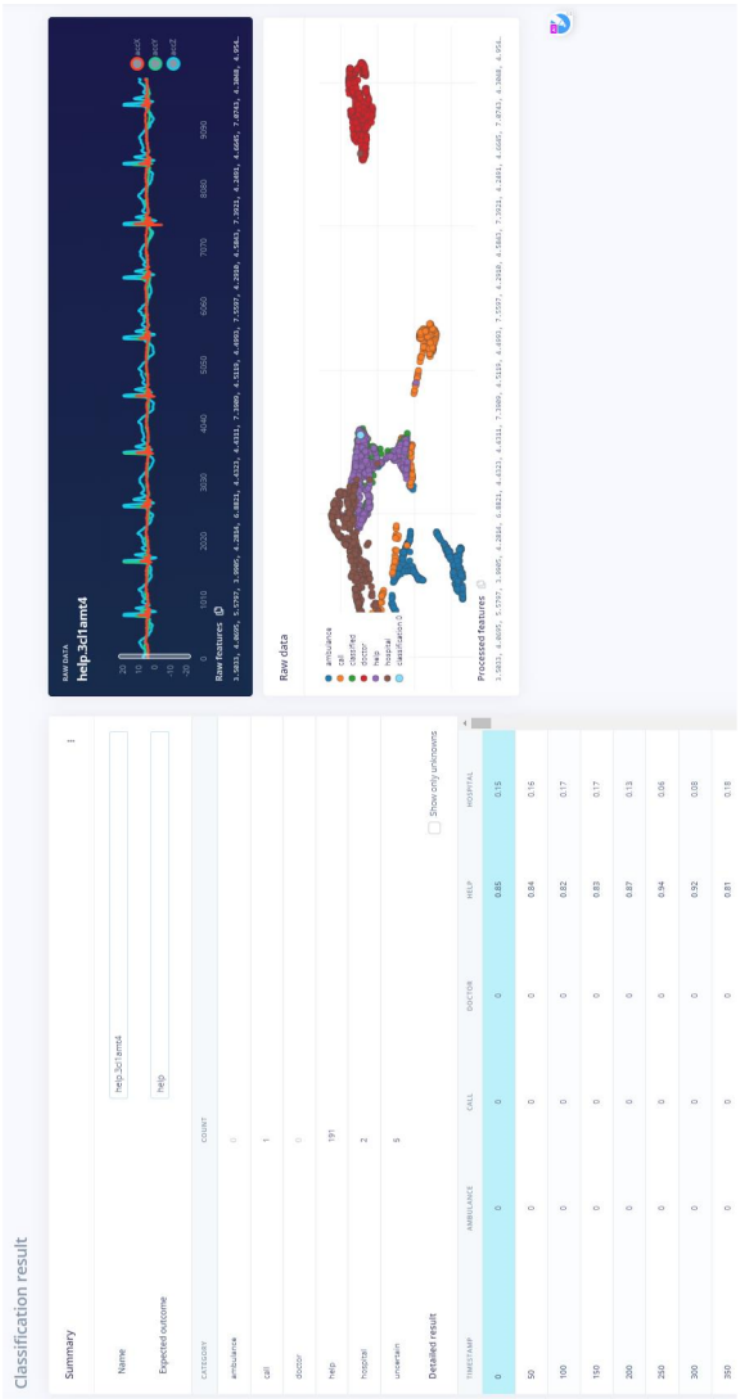


Figure 15. Sample of the on-device classification of "Help" word.

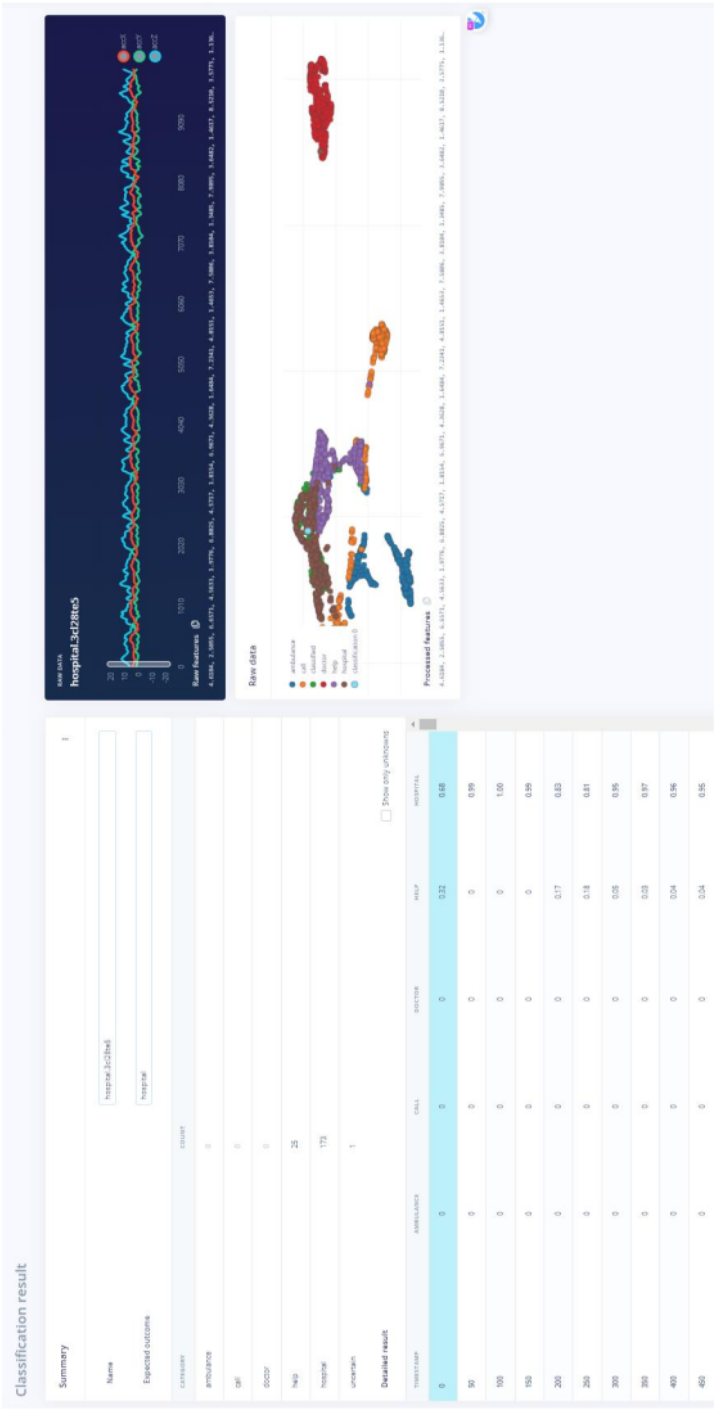


Figure 16. Sample of the on-device classification of "Hospital" word.

4.4. Analysis of the SmartCall Results

4.4.1. Offline Result

The examination of the misclassified words was executed making use of the confusion matrix (Dere et al. 2022). Figure 13(a) describes how the validation data was evaluated using the confusion matrix. The validation dataset yielded an accuracy of 91.2%, an F1-score of 0.89, and a loss of 0.26 (see Appendix II and Appendix III). Upon analysis, it was discovered that the "help" category had a lower F1-score of 0.85 but still managed an accuracy of 92.1%. However, an interesting finding was observed, specifically that the "help" category had a relatively high confusion rate with the "hospital" category, with a misclassification rate of 14.8%. This suggests that the model may require optimization to improve its ability to distinguish between these two categories in real-world scenarios. The confusion matrix of the test dataset is shown in Figure 13(b) yielding an accuracy of 89.20% on the quantized model. It is worth noting that the benefits of quantization were further illustrated in (see Figure 14), as the quantized model demonstrated a latency of only 2ms, whereas the unquantized model showed a latency of 18ms. This suggests that the quantized model is over 80% more efficient in real-time latency than the unquantized model.

4.4.2. On-device Accuracy

The on-device evaluation for experiments two and three in this thesis an identical preprocessing step as the offline experiment one. The evaluation process was carried out on the already preprocessed raw IMU data using Equation 9, which yielded an overall average accuracy of 89.20% (see Appendix IV) across all words in experiment two and 89.65% in experiment three (see Appendix VII), without a majority vote (Dere et al. 2022). However, the "help" was often confused with “hospital” or “call” which ended up resulting in the lowest accuracy (see

Appendix V and Appendix VI) even though the actual reality was among the top 3 estimated results.

Latency is a crucial parameter to consider for real-time gesture recognition devices. An ideal device should have minimal electromechanical delay to ensure effectiveness. In this study, the SmartCall design yielded a latency of 2 ms using Arm Cortex-M4F. Additionally, the peak RAM usage and 28.6K flash usage were recorded (see Figure 14.)

4.5. Discussion of the SmartCall Results

The study introduces a novel approach for real-time online classification of ASL words using deep learning. What sets this approach apart from previous studies is its utilization of raw IMU data without feature engineering for a domain-specific application. The study's findings demonstrate the efficient and accurate classification of raw input data (Dere et al. 2022).

The primary objective of this research was to facilitate communication for individuals with speech impairments during medical emergencies. The study showcased the accurate recognition of ASL words using a 1D-CNN approach and successfully deployed the model to a low-cost embedded device with an impressively low inference latency of 2 ms. Furthermore, the deployment of the model incurred minimal carbon emission costs, establishing it as an environmentally friendly solution (Dere et al. 2022).

The design cost of the device is significantly lower than that of similar devices (Refer Table 3), indicating its potential for rapid prototyping in middle and low-resource countries. Additionally, the onboard sensor ensures a compact and user-friendly design with minimal calibration requirements (Dere et al. 2022).

4.6. Benchmark of the SmartCall Results

It is almost impossible to compare results because the aim of the study is to novel without any prior study targeting medical emergencies words. However, in Table 3 we try to compare the offline results with application specific SLR literatures. The cost was estimated based on the selling price of the inference device and sensor at the time of experimentation.

Table 3. Comparison of SmartCall with other works

	(Wen et al. 2021)	(Venugopalan and Reghunadhan 2021)	**This study**
Number of words	50	13	**5**
Deployment	PC	Not applicable	**Arduino**
Estimated Cost (Naira)	> 50,000	Not applicable	**10,000**
Architecture	CNN	BiLSTM	**CNN**
Offline Accuracy (%)	86.67	76.21	**91.20**

4.7. SmartCom System Evaluation

BLEU score was used to evaluate the quality of machine-translated Sign-to-text model. The BLEU score assesses how close machine-generated text is to a human-generated translation. The calculation of the BLEU score involves comparing the machine and human-generated text on a sentence-by-sentence basis, using a weighted geometric average to compute a score for each sentence, with the final BLEU score being a geometric average of the individual sentence scores. Interpreting the BLEU score can be challenging, and it's often recommended to consider the score in the context of the domain and application to determine its usefulness. If the score is low, it may indicate that the machine-generated text is not accurate or fluent enough to be of practical use. However, a high BLEU score doesn't always guarantee a translation that is free of errors or fully understandable.

The text-to-LRL model was evaluated using a qualitative survey administered to the study participants (refer to Appendix I). From the text dataset, five random videos were selected and processed using the SmartCom model, visualizing the final output on a PC. In the first experiment, the LRL translated text was presented to the participants, who were then asked to complete Section 2 of the survey. For the second experiment, participants were shown the original text generated by the Sign-to-text model and the original transcript of the video (i.e., Ground-truth), after which they were instructed to fill out Section 3 of the survey form.

4.8. SmartCom Result Presentation

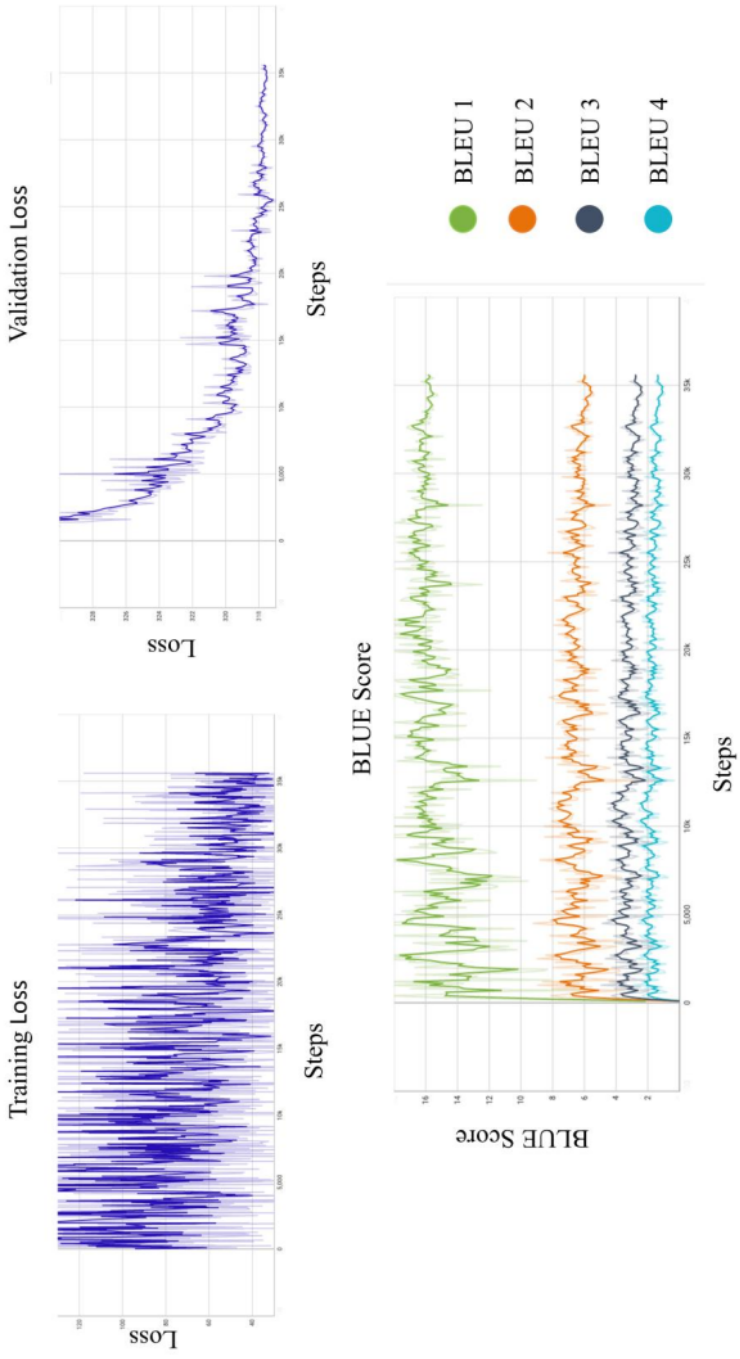


Figure 17. Model 1 Performance

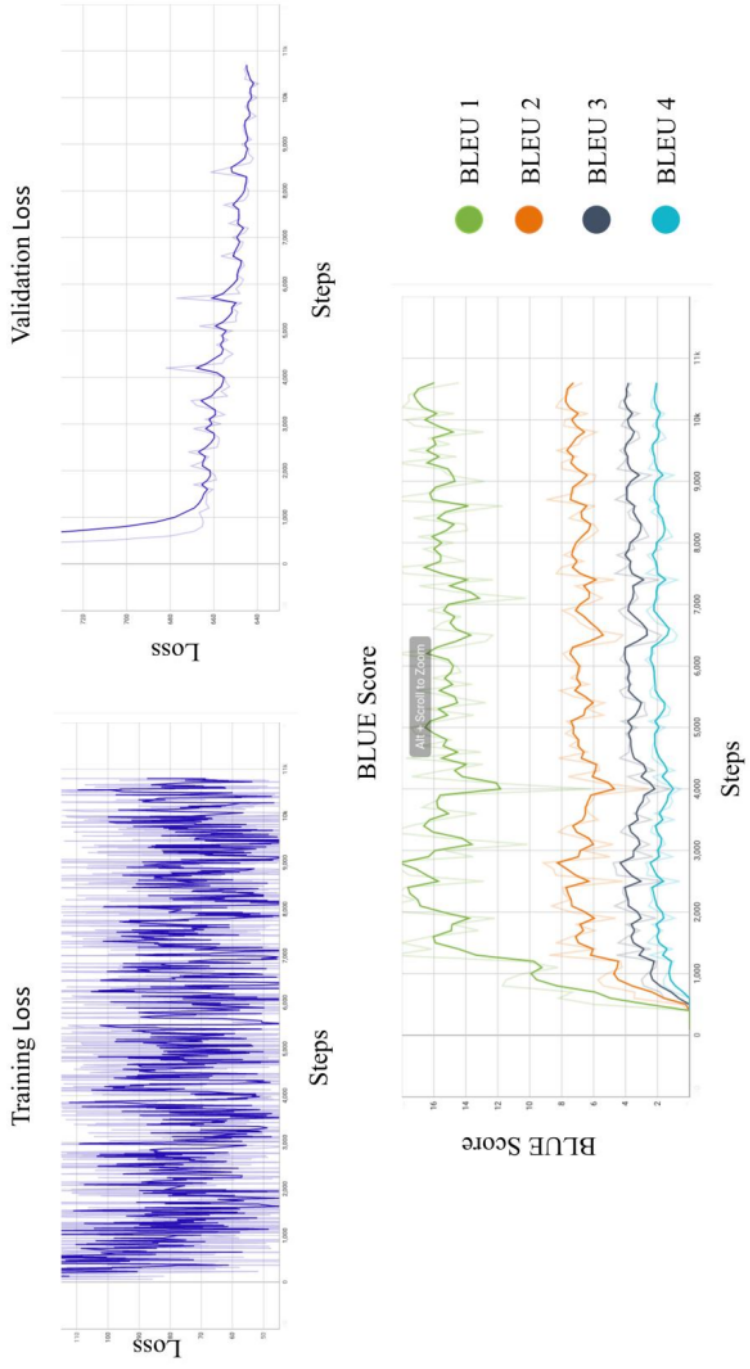


Figure 18. Model 2 Performance

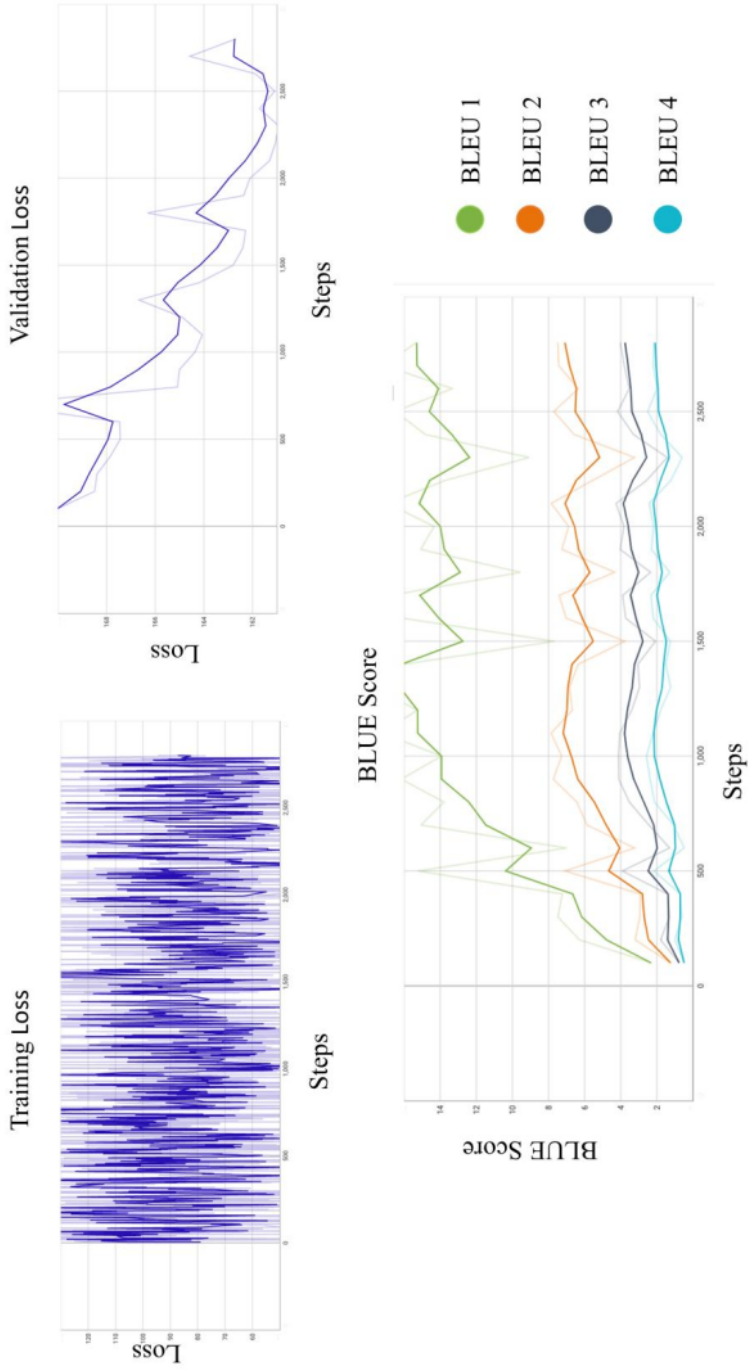


Figure 19. Model 3 Performance

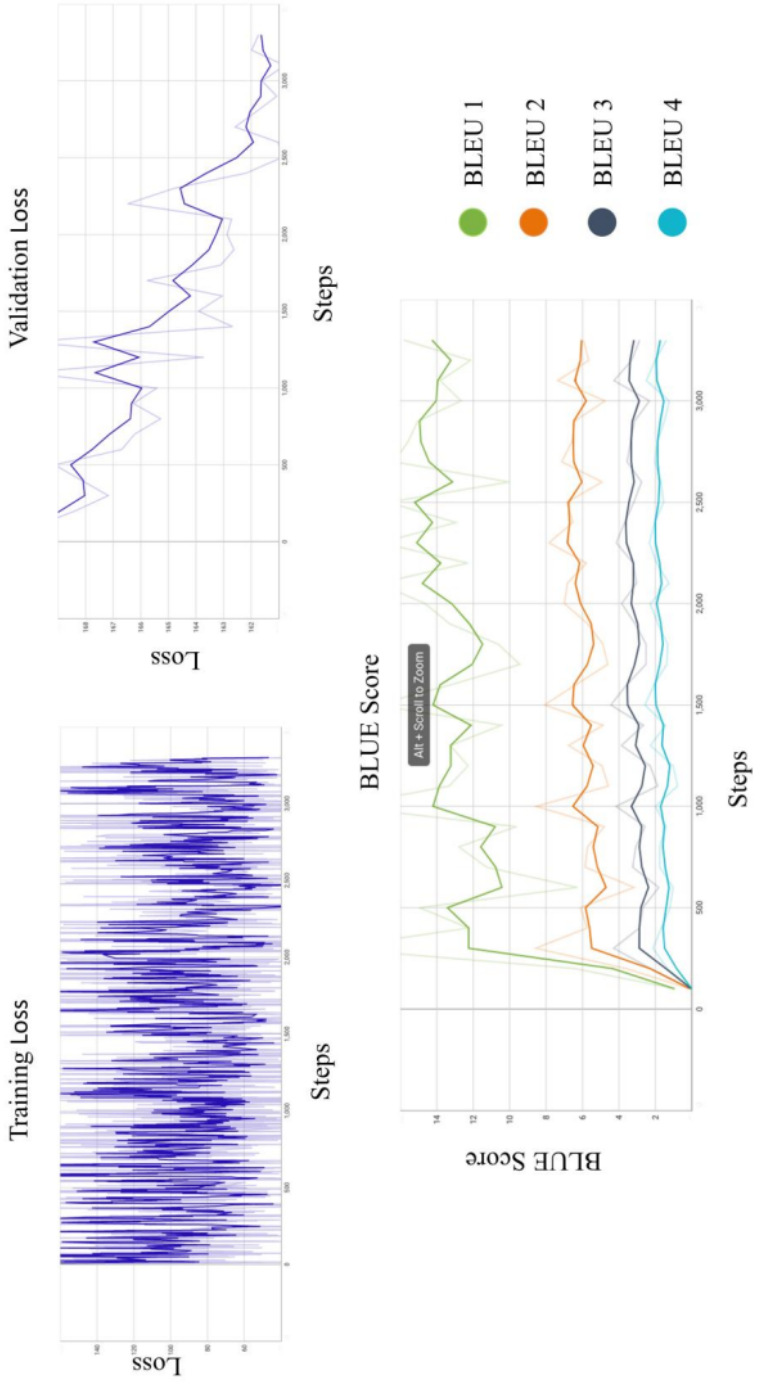


Figure 20. Model 4 Performance

	BLEU 1	BLEU 2	BLEU 3	BLEU 4
Model 1	**18.55**	**8.99**	**4.99**	**2.96**
Model 2	15.56	7.59	4.35	2.77
Model 3	13.70	7.13	3.95	2.43
Model 4	17.33	8.08	4.44	2.57

Table 4. Best BLEU scores across all the models

```
*******************************************************************
Trial 5
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Sign to Text for extraction video 8kAWy2YodzQ_12-1-rgb_front in test set
------------------------------------------------------------------------
------------------------------------------------------------------------
------------------------------------------------------------------------
------------------------------------------------------------------------
Orginal Sentence: That's pretty good.
*******************************************************************
Sentence generated by Sign2Text Model: THAT BE PRETTY GOOD
*******************************************************************
The translated sign in Hausa is: Wannan yana da kyau.
*******************************************************************
End
```

Figure 21. Translation of sentence five to Hausa

```
**********************************************************************
Trial 5
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Sign to Text for extraction video 8kAWy2YodzQ_12-1-rgb_front in test set
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
Orginal Sentence: That's pretty good.
**********************************************************************
Sentence generated by Sign2Text Model: THAT BE PRETTY GOOD
**********************************************************************
The translated sign in Ibo is: Nke ahụ dị nnọọ mma.
**********************************************************************
End
```

Figure 22. Translation of sentence five to Ibo

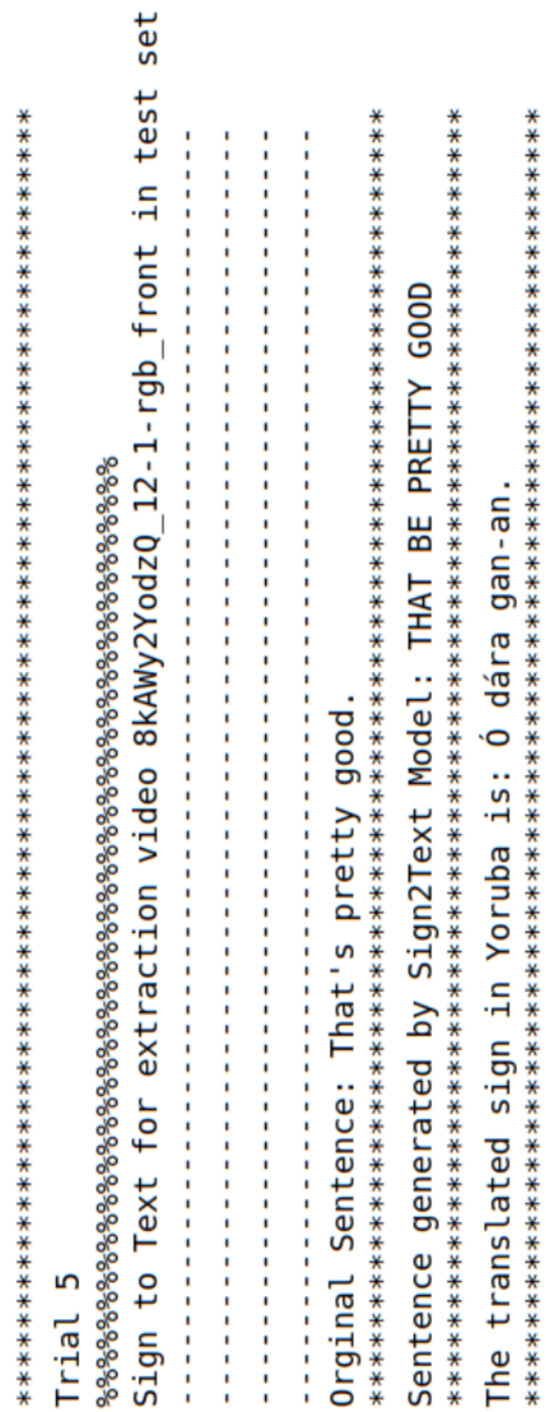


Figure 23. Translation of sentence five to Yoruba

4.9. Analysis of the SmartCom Results

4.9.1. Quantitative Analysis

In the analysis of the performance of different models, several key observations were made. Stable training and validation loss were consistently observed in Model 1 across all steps, as evidenced in Figure 17. This stability in loss indicates a consistent and reliable performance of the model throughout the training process. Furthermore, it was noted that increasing the number of heads and layers in the model, particularly when using a small dataset, led to sub-optimal BLEU scores. This is clearly depicted in Figure 19, highlighting the sensitivity of model performance to its configuration, especially in the context of limited data. The use of the plateau scheduling method was found to have a significant impact on the model's ability to learn global context. This hindrance is illustrated in both Figure 19 and Figure 20, indicating the potential limitations of this scheduling approach in capturing broader linguistic contexts (Dere et al. 2023).

A comparative analysis, as presented in Table 4, specifically focusing on BLEU-4 scores, revealed that Model 3 and Model 4, both utilizing the plateau scheduling method, resulted in the two lowest BLEU scores. This comparison underscores the potential drawbacks of employing this scheduling method in training, particularly in the context of achieving higher linguistic accuracy and fluency. The analysis of model performance and scheduling methods training highlights the critical influence of model configuration and scheduling techniques on the overall effectiveness of language processing models. These findings underscore the importance of carefully considering these factors in the development and optimization of the models for various applications (Dere et al. 2023).

In the assessment of translation quality, BLEU scores are utilized to measure the fluency and adequacy of translations based on different n-gram values. Specifically, unigram BLEU scores (n-gram value of 1) are employed to assess fluency, while higher n-gram values such as bigrams (n-gram value of 2) and trigrams (n-gram value of 3) are utilized to evaluate the adequacy of translations by identifying relevant word sequences, particularly in specialized domains with specific terminologies. Moreover, the use of an n-gram value of 4 places greater emphasis on precise matches, making it beneficial for evaluating highly specific or technical translations. The BLEU scores for all models examined in the present study have been documented in Table 4. Notably, the focus of this investigation centers on the BLEU-4 scores, specifically in relation to translation sequence matching (Dere et al. 2023).

The performance of Model 1 in terms of BLEU-1, BLEU-2, BLEU-3, and BLEU-4 is detailed in Table 4, revealing higher scores compared to other models. These findings indicate the effectiveness of the Transformer model in facilitating the mapping between sign words, gloss, and generated words. This is particularly evident in the notably high BLEU-4 score, signifying precise matches in the gestured words. Overall, the results underscore the efficacy of the Transformer model in accurately translating sign language, as evidenced by the BLEU scores (Dere et al. 2023).

4.9.2. Qualitative Analysis

Providing insight into the end-to-end model's effectiveness for translating sentence five in low-resource languages is investigated using qualitative analysis. Figure 21, Figure 22, and Figure 23 demonstrate that the sign-to-text model did not generate the original sentence with perfect accuracy. Despite this, the generated text was able to convey the sentence's meaning

successfully. These results emphasize the model's limitations for generating text from sign language but suggest that it still has potential to generate useful translations in these contexts.

4.10. Discussion of the SmartCom Results

The present study has implemented an end-to-end framework specifically designed to facilitate the translation of ASL into LRL such as Hausa, Ibo, and Yoruba, which are widely communicated across Nigeria. The successful execution of this model holds the potential to have a profound impact on the integration of individuals with hearing impairments into various mainstream societal domains, including education and media. Through the utilization of this framework, there exists the potential to effectively bridge the communication gap between the DHH community and the broader population, which predominantly employs spoken languages for communication purposes. The findings of this study lay the groundwork for future research endeavors and the development of analogous tools aimed at enhancing accessibility for individuals with disabilities in diverse contexts. These advancements are poised to substantially diminish disparities and foster the creation of a more inclusive society, thereby ensuring that no individual is marginalized in terms of education, employment, and other critical domains (Dere et al. 2023).

In this study, the Transformer model demonstrated robustness in using video data to automatically generate words based on the contextual information provided by the gloss. The CNN network facilitated the video features extraction process. Nevertheless, it is important to note that the inherent limitation of CNN in modeling long-range video dependencies represents a notable drawback. Subsequent research endeavors to advance the limitation by integrating a pretrained vision Transformer model to facilitate feature extraction from the video, thereby extending the model's capacity to comprehend dependencies between sign gestures within a given context. Furthermore, the gloss utilized in this study was exclusively generated by means of a statistical model, as the original How2Sign dataset did not encompass gloss information.

Consequently, the model exhibited suboptimal performance in this aspect. As a prospective avenue for further exploration, the integration of a meticulously curated gloss holds the potential to significantly enhance the performance of the text generation Transformer (Dere et al. 2023).

The NLLB model successfully developed translations that were readily understandable to native speakers of the LRL. However, it is important to note that the model rely on the identification of the closest matching word than considering the context of sentence Furthermore, the accurate phonemic annotation of languages that utilize them are not provided by the NLLB model, thereby potentially presenting challenges in both the composition and comprehension of the sentences. Despite these limitations, every participant involved in the study expressed high levels of satisfaction with the performance of both the sign-to-text and text-to-LRL models. These encouraging outcomes underscore the potential for further research and enhancements aimed at augmenting the overall performance of the model (Dere et al. 2023).

While the proposed framework is innovative, its evaluation was conducted using non-native NSL speakers, potentially restricting its applicability to real-world scenarios. Accordingly, forthcoming research endeavors intend to engage native NSL speakers to assess the framework's efficacy in facilitating the transmission of instructional information within an educational context. Notwithstanding this limitation, the encouraging findings indicate that the proposed framework holds significant promise in bridging communication divides between individuals with hearing and speech impairments and the broader population. As a result, it has the potential to enhance accessibility to mainstream facets of the economy, including education and media (Dere et al. 2023).

4.11. Benchmark of the Results

	(Duarte et al. 2021)	(Pillai and Pietri 2022)	**This work**
Dataset	How2Sign	How2Sign	**How2Sign**
BLEU-1	12.28	19.72	**18.55**
BLEU-2	6.71	5.76	**8.99**
BLEU-3	3.32	2.30	**4.99**
BLEU-4	1.89	1.02	**2.96**

CHAPTER 5

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1. Summary

The summary is structured with two parts - the first part provides a summary of SmartCall, an AI-solution that enables DHH communicate during medical emergencies, while the second part provides a summary of SmartCom, an AI-framework that enables seamlessly communication in selected languages in Nigeria.

5.1.1. SmartCall Summary

The first solution in this thesis proposes an innovative 1D-CNN model that is deployed on an embedded device worn around the wrist. Its primary function is to enable seamless communication between individuals with speaking impairments and others, regardless of whether they have speech or hearing impairments, in medical emergency situations. The evaluation of the model's offline and on-device accuracy justifies its effectiveness and optimization technique.

5.1.2. SmartCom Summary

The second solution in this thesis proposes a framework that can potentially facilitate communication connections and enhance attainability for individuals with hearing and speech impairments in various mainstream aspects, including the economy, education, and media. Through the proposed frameworks, these individuals can participate meaningfully in economic and social activities, potentially improving their quality of life and promoting inclusivity.

5.2. Conclusion

This thesis introduces two solutions towards leveraging AI as a tool to bridge communication gap for DHH community and possibly integrating them as instructors in the education section in Nigeria while also providing a real-time solution to help them communicate during period of medical emergencies.

In conclusion, this thesis proposes a solution called SmartCall that aims to address challenges in communication faced by individuals with hearing or speech impairments in medical emergency situations. The wristwatch like design can recognize five words from DHH signage using a deep learning algorithm deployed on an embedded device. The offline and on-device evaluation showed an accuracy of more than 80% with a low latency of 2 ms during inference, demonstrating the potential of the proposed solution.

Additionally, this thesis proposes an innovative solution called SmartCom that aims to bridge communication gaps faced by individuals with hearing or speech impairments, particularly in educational settings. By using a camera sensor to capture sign language gestures and translating them into indigenous languages in Nigeria, the SmartCom presents a framework that can enable meaningful participation of individuals with DHH in educational activities. The study yielded high performance in both the Sign-to-text and text-to-LRL models, demonstrating the potential of the proposed solution.

5.3. Recommendations

Based on the findings of this thesis, we recommend the adoption of both the SmartCall and SmartCom solutions to promote effective communication and inclusivity for individuals with hearing or speech impairments in medical emergencies and educational settings. The SmartCall device showed high accuracy in recognizing key emergency words with a low latency, while the SmartCom framework demonstrated promising performance in translating sign language to indigenous languages, enabling meaningful participation in educational activities. We believe that the adoption of these solutions could significantly improve communication access and enhance the quality of life for individuals with hearing or speech impairments.

5.4. Contributions to Knowledge

As a contribution, the thesis highlights a novel approach proposed for seamless communication for DHH individuals in a domain specific application. The approach uses AI to translate ASL to three Nigerian languages: Hausa, Ibo, and Yoruba. The model weights are quantized and saved, which provides a potential for patenting or making it available as an API to the public. The proposed approach in the thesis can inspire further research on the development of accessible technologies that can improve quality of life and promote inclusivity for individuals with disabilities in diverse sociocultural contexts.

This thesis can be useful for further research and development of assistive technologies in the field of disability studies and can inspire scholars to work towards creating a more inclusive and accessible society. As a result, it can contribute to the development of policies, practices, and technologies that promote equal opportunities for individuals with disabilities and social equity.

5.5. Future Research Directions

One significant recommendation for the SmarCall solution is to include a communication phone for audio speech output and text visualization of the inference words to improve the practicality of SmartCall. Moreover, the word in the dataset is limited, increasing the number of emergency words can significantly improve the practicality of SmartCall.

One potential avenue for future work is to enhance the BLEU score of the How2Sign dataset. Additionally, validating the efficacy and practicality of the SmartCom framework in a classroom environment with individuals who have DHH impairments could serve as further evidence of its effectiveness.

References

Abdul, Wadood, Mansour Alsulaiman, Syed Umar, Mohammed Faisal, Ghulam Muhammad, Fahad R. Albogamy, Mohamed A. Bencherif, and Hamid Ghaleb. 2021. "Intelligent Real-Time Arabic Sign Language Classification Using Attention-Based Inception and BiLSTM ☆." *Computers and Electrical Engineering* 95(September 2020):107395. doi: 10.1016/j.compeleceng.2021.107395.

Al-Hammadi, Muneer, Ghulam Muhammad, Wadood Abdul, Mansour Alsulaiman, Mohamed A. Bencherif, and Mohamed Amine Mekhtiche. 2020. "Hand Gesture Recognition for Sign Language Using 3DCNN." *IEEE Access* 8:79491–509. doi: 10.1109/ACCESS.2020.2990434.

Alwarfalli, Abdelfattah Bushnaf, Fathiyah Hamid Yousuf, and Ibrahim S. Ighneiwa. 2021. "Arabic Sign Language Recognition System by Using Surface Intelligent EMG Signal." *ACM International Conference Proceeding Series* 2–7. doi: 10.1145/3492547.3492606.

Arduino Official Store. 2020. "Arduino Nano 33 BLE — Arduino Official Store." *Store.Arduino.Cc/Usa/*. Retrieved August 18, 2022 (https://store.arduino.cc/products/arduino-nano-33-ble).

Arvanitis, Nikolaos, Constantinos Constantinopoulos, and Dimitrios Kosmopoulos. 2019. "Translation of Sign Language Glosses to Text Using Sequence-to-Sequence Attention Models." *Proceedings - 15th International Conference on Signal Image Technology and Internet Based Systems, SISITS 2019* 296–302. doi: 10.1109/SITIS.2019.00056.

Asonye, Emmanuel I. 2022. "Bridging Language Gap, Promoting Deaf Literacy in Nigeria Through Indigenous Sign Languages." Pp. 285–302 in *Current Issues in Descriptive Linguistics and Digital Humanities: A Festschrift in Honor of Professor Eno-Abasi Essien Urua*. Springer.

ASONYE, Emmanuel Ihechi, and Mary EDWARD. 2022. "Deaf Education and Signed Language Situation in Ghana and Nigeria: Six Decades after Andrew Foster." *Signed Languages, Interpreting, and the Deaf Community in Ghana*.

Asonye, Emmanuel Ihechi, Ezinne Emma-Asonye, and Mary Edward. 2018. "Deaf in Nigeria: A Preliminary Survey of Isolated Deaf Communities." *SAGE Open* 8(2). doi: 10.1177/2158244018786538.

Barbhuiya, Abul Abbas, Ram Kumar Karsh, and Rahul Jain. 2021. "CNN Based Feature Extraction and Classification for Sign Language." *Multimedia Tools and Applications* 80(2):3051–69. doi: 10.1007/s11042-020-09829-y.

Blench, Roger, Andy Warren, and Mallam Dendo UBS. 2006. "An Unreported African Sign Language for the Deaf among the Bura in Northeast Nigeria." in *Unpublished conference handout*.

Carreira, João, and Andrew Zisserman. 2017. “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset.” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017* 2017-Janua:4724–33. doi: 10.1109/CVPR.2017.502.

Chai, Junyi, Hao Zeng, Anming Li, and Eric W. T. Ngai. 2021. “Deep Learning in Computer Vision: A Critical Review of Emerging Techniques and Application Scenarios.” *Machine Learning with Applications* 6(August):100134. doi: 10.1016/j.mlwa.2021.100134.

Chandra, Mayank Arya, and S. S. Bedi. 2021. “Survey on SVM and Their Application in Image Classification.” *International Journal of Information Technology (Singapore)* 13(5):1867–77. doi: 10.1007/s41870-017-0080-1.

Cheok, Ming Jin, Zaid Omar, and Mohamed Hisham Jaward. 2019. “A Review of Hand Gesture and Sign Language Recognition Techniques.” *International Journal of Machine Learning and Cybernetics* 10(1):131–53.

Chollet, François. 2017. *Machine Learning*. Vol. 45. McGraw-hill New York.

Courville, Ian Goodfellow and Yoshua Bengio and Aaron. 2016. *Deep Learning*. Vol. 29. MIT press.

Deji Dere, Mustapha, Roshidat Oluwabukola Dere, Adewale Adesina, and Aliyu Rufai Yauri. 2022. “SmartCall: A Real-Time, Sign Language Medical Emergency Communicator.” *Proceedings of the 5th International Conference on Information Technology for Education and Development: Changing the Narratives Through Building a Secure Society with Disruptive Technologies, ITED 2022* 31–36. doi: 10.1109/ITED56637.2022.10051420.

Dere, Mustapha Deji, Roshidat Oluwabukola Dere, Adewale Adesina, and Aliyu Rufai Yauri. 2023. “An End-to-End Framework for Translation of American Sign Language to Low-Resource Languages in Nigeria.” *Scientific African* 21. doi: 10.1016/j.sciaf.2023.e01809.

Dere, Mustapha Deji, Roshidat Oluwabukola Dere, Adewale Adesina, and Aliyu Rufai Yauri. 2022. “SmartCall: A Real-Time, Sign Language Medical Emergency Communicator.” in *Proceedings of the 5th International Conference on Information Technology for Education and Development: Changing the Narratives Through Building a Secure Society with Disruptive Technologies, ITED 2022*. Institute of Electrical and Electronics Engineers Inc.

Devlin, Jacob, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1(Mlm):4171–86.

Dodge, Jesse, Taylor Prewitt, Remi Tachet des Combes, Erika Odmark, Roy Schwartz, Emma Strubell, Alexandra Sasha Luccioni, Noah A. Smith, Nicole DeCario, and Will Buchanan. 2022. “Measuring the Carbon Intensity of AI in Cloud Instances.” *2022 ACM Conference*

on Fairness, Accountability, and Transparency (FAccT '22), June 21â•fi24, 2022, Seoul, Republic of Korea 1(1):1877–94. doi: 10.1145/3531146.3533234.

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale."

Duarte, Amanda, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-I-Nieto. 2021. "How2Sign: A Large-Scale Multimodal Dataset for Continuous American Sign Language." *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2734–43. doi: 10.1109/CVPR46437.2021.00276.

Faisal, Md Ahasan Atick, Farhan Fuad Abir, Mosabber Uddin Ahmed, and Md Atiqur Rahman Ahad. 2022. "Exploiting Domain Transformation and Deep Learning for Hand Gesture Recognition Using a Low-Cost Dataglove." *Scientific Reports* 12(1):1–15. doi: 10.1038/s41598-022-25108-2.

Ferrite, Silvia, Islay Mactaggart, Hannah Kuper, Joseph Oye, and Sarah Polack. 2017. "Prevalence and Causes of Hearing Impairment in Fundong Health District, North-West Cameroon." *Tropical Medicine and International Health* 22(4):485–92. doi: 10.1111/tmi.12840.

Forsyth, David A., and Jean Ponce. 2002. *Computer Vision: A Modern Approach*. prentice hall professional technical reference.

Gupta, Rinki, and Arun Kumar. 2021. "Indian Sign Language Recognition Using Wearable Sensors and Multi-Label Classification." *Computers and Electrical Engineering* 90(December 2019):106898. doi: 10.1016/j.compeleceng.2020.106898.

Han, Hui, and Julien Siebert. 2022. "TinyML: A Systematic Review and Synthesis of Existing Research." *4th International Conference on Artificial Intelligence in Information and Communication, ICAIIC 2022 - Proceedings* 269–74.

Haque, Promila, Badhon Das, and Nazmun Nahar Kaspy. 2019. "Two-Handed Bangla Sign Language Recognition Using Principal Component Analysis (PCA) and KNN Algorithm." *2nd International Conference on Electrical, Computer and Communication Engineering, ECCE 2019* 7–9. doi: 10.1109/ECACE.2019.8679185.

Hassan, S. T., J. A. Abolarinwa, C. O. Alenoghena, S. A. Bala, M. David, and Ali Farzaminia. 2017. "Intelligent Sign Language Recognition Using Enhanced Fourier Descriptor: A Case of Hausa Sign Language." *Proceedings - 2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems, I2CACIS 2017* 2017-Decem(October):104–9. doi: 10.1109/I2CACIS.2017.8239041.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. “Deep Residual Learning for Image Recognition.” Pp. 770–78 in *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Horning, N. 2010. “Random Forests: An Algorithm for Image Classification and Generation of Continuous Fields Data Sets.” *International Conference on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences 2010* 1–6.

Hymel, Shawn, Colby Banbury, Daniel Situnayake, Alex Elium, Carl Ward, Mat Kelcey, Mathijs Baaijens, Mateusz Majchrzycki, Jenny Plunkett, David Tischler, Alessandro Grande, Louis Moreau, Dmitry Maslov, Artie Beavis, Jan Jongboom, and Vijay Janapa Reddi. 2022. “Edge Impulse: An MLOps Platform for Tiny Machine Learning.”

Iqbal, Touseef, and Shaima Qureshi. 2022. “The Survey: Text Generation Models in Deep Learning.” *Journal of King Saud University - Computer and Information Sciences* 34(6):2515–28. doi: 10.1016/j.jksuci.2020.04.001.

Janapa Reddi, Vijay, Brian Plancher, Susan Kennedy, Laurence Moroney, Pete Warden, Lara Suzuki, Anant Agarwal, Colby Banbury, Massimo Banzi, Matthew Bennett, Benjamin Brown, Sharad Chitlangia, Radhika Ghosal, Sarah Grafman, Rupert Jaeger, Srivatsan Krishnan, Maximilian Lam, Daniel Leiker, Cara Mann, Mark Mazumder, Dominic Pajak, Dhilan Ramaprasad, J. Evan Smith, Matthew Stewart, and Dustin Tingley. 2022. “Widening Access to Applied Machine Learning with TinyML.” *Harvard Data Science Review* 1–20. doi: 10.1162/99608f92.762d171a.

Kahlon, Navroz Kaur, and Williamjeet Singh. 2021. *Machine Translation from Text to Sign Language: A Systematic Review*. Vol. 22. Springer Berlin Heidelberg.

Khomami, Sara Askari, and Sina Shamekhi. 2021. “Persian Sign Language Recognition Using IMU and Surface EMG Sensors.” *Measurement: Journal of the International Measurement Confederation* 168(August 2020):108471. doi: 10.1016/j.measurement.2020.108471.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. 2017. “ImageNet Classification with Deep Convolutional Neural Networks.” *Communications of the ACM* 60(6):84–90. doi: 10.1145/3065386.

Leon, Vasileios, Spyridon Mouselinos, Konstantina Koliogeorgi, Sotirios Xydis, Dimitrios Soudris, and Kiamal Pekmestzi. 2020. “A TensorFlow Extension Framework for Optimized Generation of Hardware CNN Inference Engines.” *Technologies* 8(1):6. doi: 10.3390/technologies8010006.

Liu, Yilin, Fengyang Jiang, and Mahanth Gowda. 2020. “Finger Gesture Tracking for Interactive Applications: A Pilot Study with Sign Languages.” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4(3). doi: 10.1145/3414117.

Magdy, Mostafa, Balaha Sara, El-kady Hossam Magdy, Mohamed Salama, Eslam Emad, Muhammed Hassan, and Mahmoud M. Saafan. 2022. "A Vision-Based Deep Learning Approach for Independent-Users Arabic Sign Language Interpretation."

Mulwafu, W., H. Kuper, and R. J. H. Ensink. 2016. "Prevalence and Causes of Hearing Impairment in Africa." *Tropical Medicine and International Health* 21(2):158–65. doi: 10.1111/tmi.12640.

Occhipinti, Annalisa, Louis Rogers, and Claudio Angione. 2022. "A Pipeline and Comparative Study of 12 Machine Learning Models for Text Classification." *Expert Systems with Applications* 201(April):117193. doi: 10.1016/j.eswa.2022.117193.

Olabanji, Ayodele Olawale, and Akinlolu Adediran Ponnle. 2021. "Development of A Computer Aided Real-Time Interpretation System for Indigenous Sign Language in Nigeria Using Convolutional Neural Network." *European Journal of Electrical Engineering and Computer Science* 5(3):68–74. doi: 10.24018/ejece.2021.5.3.332.

Othman, Achraf, and Mohamed Jemni. 2019. "Designing High Accuracy Statistical Machine Translation for Sign Language Using Parallel Corpus: Case Study English and American Sign Language." *Journal of Information Technology Research (JITR)* 12(2):134–58.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei Jing Zhu. 2002. "BLEU: A Method for Automatic Evaluation of Machine Translation." Pp. 311–18 in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. Vols. 2002-July.

Patterson, David, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. "Carbon Emissions and Large Neural Network Training." 1–22.

Pereira-Montiel, E., E. Pérez-Giraldo, J. Mazo, D. Orrego-Metaute, E. Delgado-Trejos, D. Cuesta-Frau, and J. Murillo-Escobar. 2022. "Automatic Sign Language Recognition Based on Accelerometry and Surface Electromyography Signals: A Study for Colombian Sign Language." *Biomedical Signal Processing and Control* 71(November 2020). doi: 10.1016/j.bspc.2021.103201.

Pillai, Nitin, and Isabel Garcia Pietri. 2022. "MyVoice : Continuous End-to-End Sign Language to Text Translation." *Unpublished*.

Quinn, M., and J. I. Olszewska. 2019. "British Sign Language Recognition in the Wild Based on Multi-Class SVM." *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, FedCSIS 2019* 18:81–86. doi: 10.15439/2019F274.

Al Rashid Agha, Rawan A., Muhammed N. Sefer, and Polla Fattah. 2018. "A Comprehensive Study on Sign Languages Recognition Systems Using (SVM, KNN, CNN and ANN)." *ACM International Conference Proceeding Series*. doi: 10.1145/3279996.3280024.

Sasu, D. 2022. "Nigeria: Languages by Number of Speakers 2021 | Statista." *Languages in Nigeria 2021, by Number of Speakers*. Retrieved August 18, 2022 (https://www-statista-

com.eu1.proxy.openathens.net/statistics/1285383/population-in-nigeria-by-languages-spoken/).

Shafique, Muhammad, Theocharis Theocharides, Vijay Janapa Reddy, and Boris Murmann. 2021. "TinyML: Current Progress, Research Challenges, and Future Roadmap." *Proceedings - Design Automation Conference* 2021-Decem:1303–6. doi: 10.1109/DAC18074.2021.9586232.

Sharma, Sakshi, and Sukhwinder Singh. 2021. "Vision-Based Hand Gesture Recognition Using Deep Learning for the Interpretation of Sign Language." *Expert Systems with Applications* 182(February):115657. doi: 10.1016/j.eswa.2021.115657.

Sharma, Shikhar, and Krishan Kumar. 2021. "ASL-3DCNN: American Sign Language Recognition Technique Using 3-D Convolutional Neural Networks." *Multimedia Tools and Applications* 80(17):26319–31. doi: 10.1007/s11042-021-10768-5.

Smaira, Lucas, João Carreira, Eric Noland, Ellen Clancy, Amy Wu, and Andrew Zisserman. 2020. "A Short Note on the Kinetics-700-2020 Human Action Dataset." (i).

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." *Advances in Neural Information Processing Systems* 2017-Decem(Nips):5999–6009.

Venugopalan, Adithya, and Rajesh Reghunadhan. 2021. "Applying Deep Neural Networks for the Automatic Recognition of Sign Language Words: A Communication Aid to Deaf Agriculturists." *Expert Systems with Applications* 185(September 2020):115601. doi: 10.1016/j.eswa.2021.115601.

Wen, Feng, Zixuan Zhang, Tianyiyi He, and Chengkuo Lee. 2021. "AI Enabled Sign Language Recognition and VR Space Bidirectional Communication Using Triboelectric Smart Glove." *Nature Communications* 12(1):1–13. doi: 10.1038/s41467-021-25637-w.

Zhang, Qian, Run Zhao, Dong Wang, and Yinggang Yu. 2019. "MyoSign: Enabling End-to-End Sign Language Recognition with Wearables." *International Conference on Intelligent User Interfaces, Proceedings IUI* Part F1476:650–60. doi: 10.1145/3301275.3302296.

Zhou, Zhihao, Kyle Chen, Xiaoshi Li, Songlin Zhang, Yufen Wu, Yihao Zhou, Keyu Meng, Chenchen Sun, Qiang He, Wenjing Fan, Endong Fan, Zhiwei Lin, Xulong Tan, Weili Deng, Jin Yang, and Jun Chen. 2020. "Sign-to-Speech Translation Using Machine-Learning-Assisted Stretchable Sensor Arrays." *Nature Electronics* 3(9):571–78. doi: 10.1038/s41928-020-0428-6.

Appendices

Appendix I

This form is prepared as part of a survey towards a PhD thesis.

Section 1

The researcher promises to maintain privacy of all information provided and use it responsibly only for research purposes.

1. Email

2. Do you agree to participate in this study? *

Mark only one oval.

Yes

No

Other:

3. Do you agree that the researcher can share the findings from this study without restrictions? *

Mark only one oval.

Yes

No

Basic Information

This is to collect basic information to understand the participant and grasp of the low resource language.

4. What is your native language? *

Mark only one oval.

Hausa

Ibo

Yoruba

5. What is your proficiency level in speaking your native language? *

Mark only one oval.

Very bad

Bad

Moderate

Good

Excellent

6. What is your proficiency level in writing in your native language? *

Mark only one oval.

Very bad

Bad

Moderate

Good

Excellent

7. What is your proficiency level in reading your native language? *

Mark only one oval.

Very bad

Bad

Moderate

Good

Excellent

8. What is your proficiency level in comprehension of your native language? *

Mark only one oval.

Very bad

Bad

Moderate

Good

Excellent

Section 2

EVALAUATION BEFORE SHOWING THE ORIGINAL SENTENCE TO THE PATICIPANT

The page is meant to evaluate the performance of the NLLB model in translating the generated words from the sign-to-text model.

9. Translate Trial 1 back to English *

10. Comprehension of Trial 1 in the low resource language

Mark only one oval.

Very Bad

Excellent

1

2

3

4

5

11. Translate Trial 2 back to English *

12. Comprehension of Trial 2 in the low resource language

Mark only one oval.

Very bad

Excellent

1

2

3

4

5

13. Translate Trial 3 back to English *

14. Comprehension of Trial 3 in the low resource language

Mark only one oval.

Very bad

Excellent

1

2

3

4

5

15. Translate Trial 4 back to English *

16. Comprehension of Trial 4 in the low resource language

Mark only one oval.

Very bad

Excellent

1

2

3

4

5

17. Translate Trial 5 back to English *

18. Comprehension of Trial 5 in the low resource language

Mark only one oval.

Very bad

Excellent

1

2

3

4

5

Section 3

EVALAUATION AFTER SHOWING THE ORIGINAL SENTENCE THE ORIGINAL TEXT

The page is meant to evaluate the performance of the NLLB model after the participant is shown the original sentence and the sentence generated by the sign2text model.

19. How satisfied are you with the quality of translation? *

Mark only one oval.

Very satisfied

Satisfied

Neither satisfied nor dissatisfied

Dissatisfied

Very dissatisfied

20. How well did the model translated from original sign gesture to America English? *

Mark only one oval.

Very bad

Bad

Moderate

Good

Excellent

Appendix II

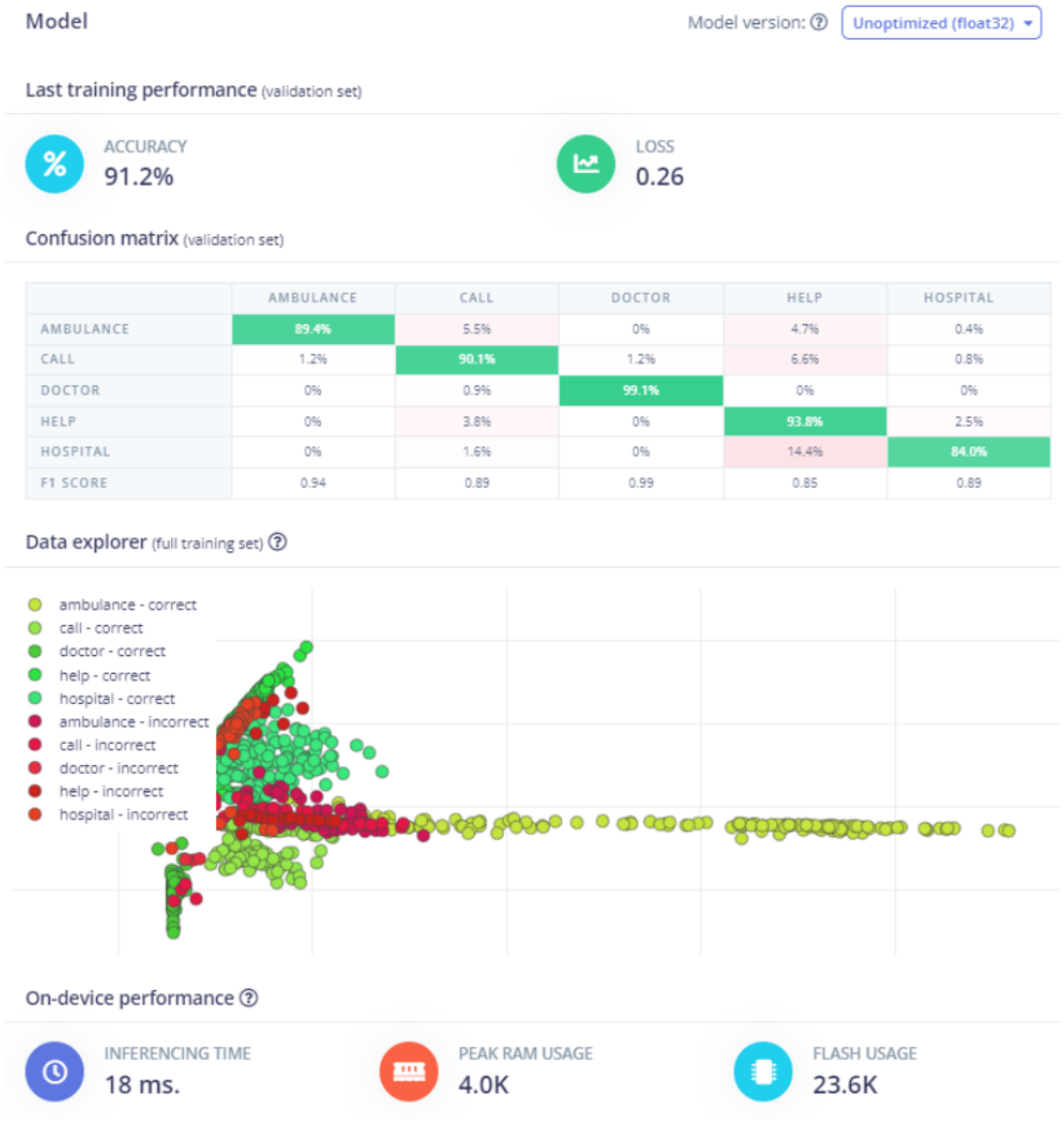


Figure 24. Unquantized 32-bit float validation result offline with on-device performance measure.

Appendix III

Figure 25. Quantized 8-bit Int validation result offline with on-device performance measure.

Appendix IV

Model testing results

ACCURACY
89.20%

	AMBULANCE	CALL	DOCTOR	HELP	HOSPITAL	UNCERTAIN
AMBULANCE	78%	5.5%	0%	3.5%	0.5%	12.5%
CALL	0%	86.5%	0%	2.5%	0.5%	10.5%
DOCTOR	0.5%	0%	99.5%	0%	0%	0%
HELP	0.5%	0.5%	0%	95.5%	1%	2.5%
HOSPITAL	0%	0%	0%	13%	86.5%	0.5%
F1 SCORE	0.87	0.90	1.00	0.89	0.92	

Feature explorer

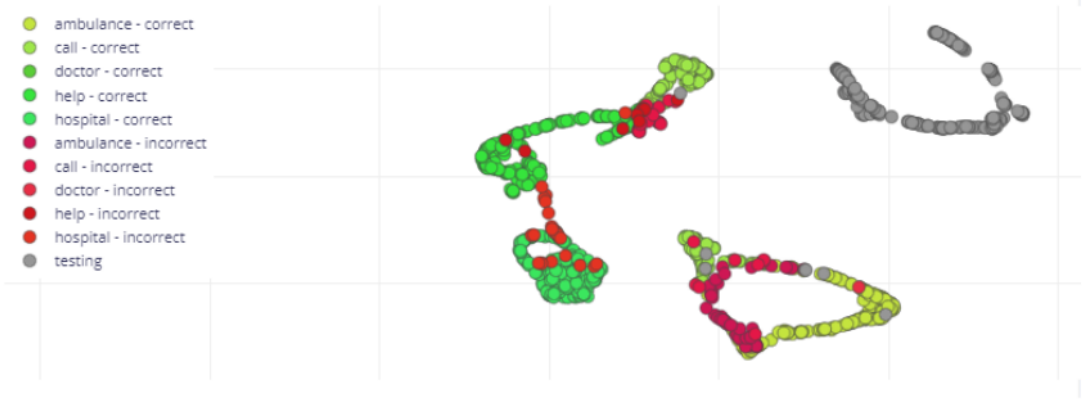


Figure 26. Test evaluation result with visualization of the word clusters.

Appendix V

Figure 27. On-device classification accuracy of "Doctor" word.

Appendix VI

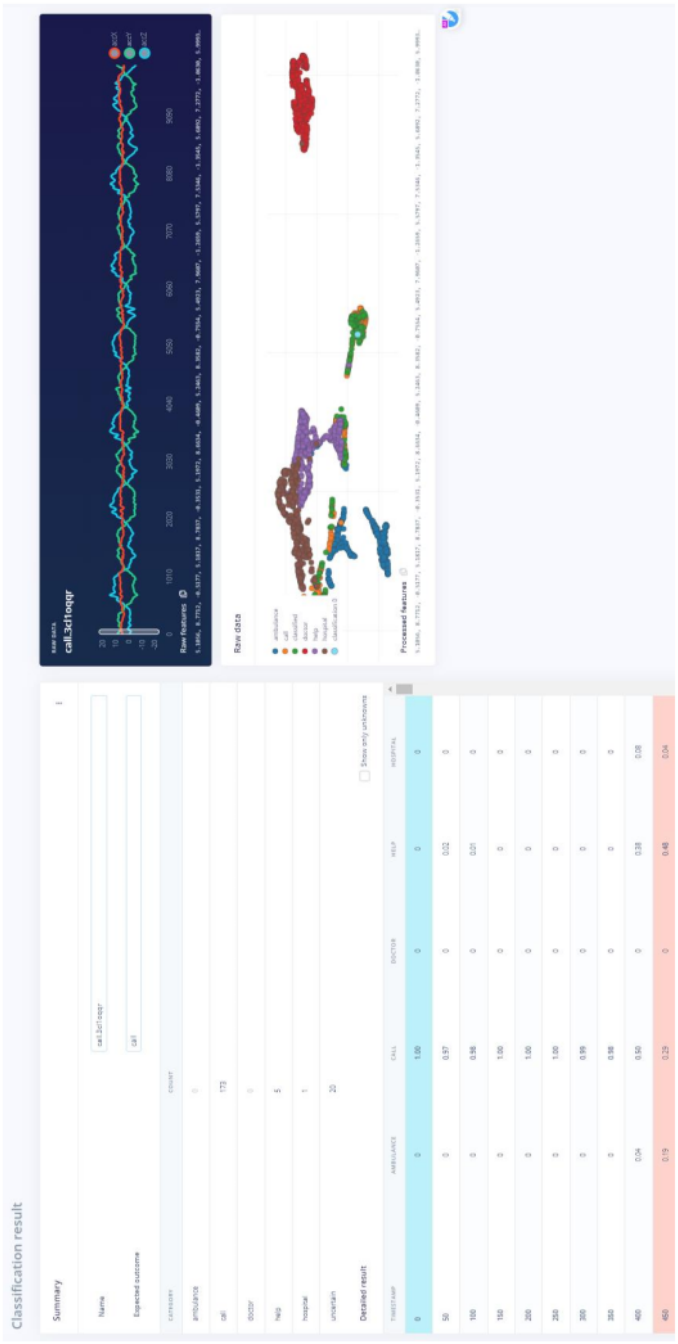


Figure 28. On-device classification accuracy of "Call" word.

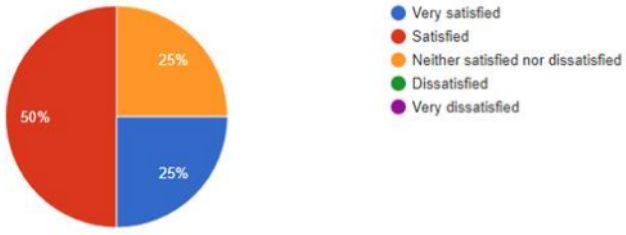
Appendix VII

Figure 29. On-device accuracy for experiment three.

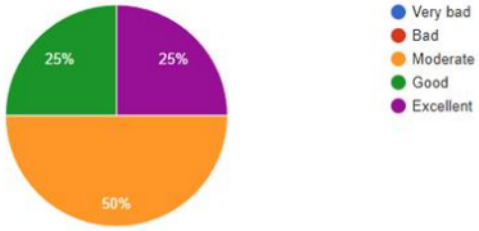
Appendix VIII

EVALAUATION AFTER SHOWING THE ORIGINAL SENTENCE THE ORIGINAL TEXT

How satisfied are you with the quality of translation ?



How well did the model translated from original sign gesture to America English?



Appendix IX

1. The entire Dere family, Offa.
2. The entire Oseni family, Offa.
3. The entire Kalejaiye family, Offa.
4. Mr. and Mrs. Lukman Oseni.
5. Mr. and Mrs. Mukalia Ijaiya.
6. Mr. and Mrs. Kehinde Oladipo.
7. Abdulrasaq Olaniyi family, Offa.
8. Offa Community.
9. University of Ibadan BME Class 2019.
10. Edge Impulse.
11. All facilitators and staff of ACETEL.

Dissertation

ORIGINALITY REPORT

13%	2%	13%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Mustapha Deji Dere, Roshidat Oluwabukola Dere, Adewale Adesina, Aliyu Rufai Yauri. "An End-to-End Framework for Translation of American Sign Language to Low-Resource Languages in Nigeria", Scientific African, 2023 Publication	9%
2	Mustapha Deji Dere, Roshidat Oluwabukola Dere, Adewale Adesina, Aliyu Rufai Yauri. "SmartCall: A Real-time, Sign Language Medical Emergency Communicator", 2022 5th Information Technology for Education and Development (ITED), 2022 Publication	2%
3	doaj.org Internet Source	1%
4	repository.smuc.edu.et Internet Source	1%

Exclude quotes	Off	Exclude matches	< 1%

Exclude bibliography On

A SUPPORT VECTOR MACHINE-BASED PROCESS FRAMEWORK FOR PREDICTING STUDENTS' ACADEMIC PERFORMANCE IN OPEN AND DISTANCE LEARNING

BY

ADEWALE, MUYIDEEN

(ACE22140007)

AFRICA CENTRE OF EXCELLENCE ON TECHNOLOGY ENHANCED LEARNING, NATIONAL OPEN UNIVERSITY OF NIGERIA

OCTOBER, 2024

A SUPPORT VECTOR MACHINE-BASED PROCESS FRAMEWORK FOR PREDICTING STUDENTS' ACADEMIC PERFORMANCE IN OPEN AND DISTANCE LEARNING

BY

ADEWALE, MUYIDEEN

(ACE22140007)

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF DOCTOR OF PHILOSOPHY (PHD) IN ARTIFICIAL INTELLIGENCE AT THE AFRICA CENTRE OF EXCELLENCE ON TECHNOLOGY ENHANCED LEARNING NATIONAL OPEN UNIVERSITY OF NIGERIA

DECLARATION

I, the undersigned **Muyideen Adewale (ACE22140007)**, hereby declare that I am the sole author of this thesis entitled **A Support Vector Machine-Based Process Framework for Predicting Students' Academic Performance in Open and Distance Learning**. To the best of my knowledge, it contains no material previously published or written by another person except where proper acknowledgement has been made. This is a true copy of the thesis, including final revisions. I acknowledge that the copyright of any published works from within the thesis resides with the respective copyright holder(s).

Date: ……

Signature: ……

CERTIFICATION/APPROVAL

This is to certify that this study was carried out by **Muyideen Adewale (ACE22140007)** at the **Africa Centre of Excellence on Technology Enhanced Learning (ACETEL), National Open University of Nigeria, Nigeria**, under my supervision.

Prof. Ambrose A. Azeta

…… ……

Main Supervisor Sign & Date

Dr Adebayo Abayomi-Alli

…… ……

Co-Supervisor Sign & Date

Dr. Amina Sambo-Magaji

…… ……

Industrial Supervisor Sign & Date

Prof. Grace E. Jokthan

…… ……

Centre Director Sign & Date

Dr. Gregory O. Onwodi

…… ……

Program Co-ordinator Sign & Date

…… ……

External Examiner Sign & Date

DEDICATION

With profound gratitude, I dedicate this thesis to the Almighty God, whose infinite mercy, grace, and love have brought me to this point. His guidance has been my constant source of strength.

This work is also dedicated to my beloved parents—my father, Mr Adewale Y. Adeleke, whose unwavering support and sacrifices have shaped my journey and to the cherished memory of my late mother, Mrs Adewale N., whose love and sacrifices continue to inspire me.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to Almighty God, the Uncreated, the Creator of all creatures. His boundless love, grace and blessings have been my constant source of strength and guidance throughout this journey.

I want to extend my most profound appreciation to my beloved parents. Despite her inability to read or write, my late mother provided unwavering support and love, instilling in me the values of perseverance and dedication. Her belief in the power of education served as a guiding light throughout my life. My father shared this belief, and together, they nurtured my aspirations from childhood to adulthood. Their sacrifices and encouragement have laid the foundation for all of my achievements.

To my siblings, Sherifdeen Kolapo Adewale, Jelilat Taiye Adewale, and Jelili Kehinde Adewale, thank you for your constant support and encouragement. Your belief in me has been a source of strength throughout this journey.

To my wonderful twins, Oluwatomisin Aamirah Adewale and Oluwadarasimi Aamir Adewale, you are my inspiration and motivation. Your presence in my life has given me the drive to overcome challenges and strive for excellence.

I am deeply grateful to my supervisors, Prof. Ambrose Azeta, Dr. Adebayo Abayomi-Alli, and Dr. Amina Sambo-Magaji, for their invaluable guidance, insightful feedback, and unwavering support. Your expertise and encouragement have been instrumental in shaping this research and bringing it to fruition.

I extend my heartfelt thanks to the Africa Centre of Excellence on Technology Enhanced Learning (ACETEL) for their financial support, which made the several international publications from this research possible. The management of ACETEL and the National Open University of Nigeria (NOUN) have provided a conducive environment for academic growth and research, for which I am deeply appreciative.

This journey would not have been possible without the support and encouragement of many individuals. I want to take this opportunity to express my deepest gratitude to all those who have contributed to the completion of this thesis. Your contributions, no matter how small, have been crucial to the completion

of this work. Thank you for being a part of my academic and personal growth.

LIST OF PUBLICATIONS

This section lists the Scopus-indexed publications that have resulted from the research conducted in this thesis. It includes papers that have been published, accepted, or are currently under review. Each publication is linked to the relevant sections of the thesis, providing a comprehensive overview of how the research contributions have been disseminated.

Title of Paper	Authors	Journal/Conference	ISSN	Thesis Sections
Impact of artificial intelligence adoption on students' academic performance in open and distance learning: A systematic literature review, **2024. Under Review.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji	Heliyon (Scopus-Indexed Journal. IF= 3.4; CiteScore=4.5).	2405-8440	Chapters 1&2
Artificial Intelligence Influence on Learner Outcomes in Distance Education: A Process-Based Framework and Research Model, **2024. Accepted.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji	EAI AFRICATEK 2024 – 7th EAI International Conference on Emerging Technologies for Developing Countries. Held in Nigeria. (Scopus-Indexed Conference Proceedings Published by Springer. CiteScore=0.7)	1867-8211	Chapter 3
A Multilayered Process Framework for Predicting Students' Academic Performance in Open and Distance Learning, **2024. Accepted.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji	EAI MTYMEX 2024 – 3rd EAI International Conference on Smart Technologies and Innovation Management. Held in Canada. (Scopus-Indexed Conference Proceedings Published by Springer. CiteScore=1.3)	2522-8609	Chapter 3
An Architectural Framework for Predicting Students' Academic Performance in Open and Distance Learning, **2024. Accepted.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji, GE Jokthan, G Onwodi, KM Lawal, and CF	EAI eLEOT 2024 – 10th EAI International Conference on e-Learning e-Education and Online Training. Held in China. (Scopus-Indexed Conference Proceedings	1867-8211	Chapter 3

	Mafiana	Published by Springer. CiteScore=0.7)		

LIST OF PUBLICATIONS (Contd.)

Empirical Investigation of Multilayered Framework for Predicting Academic Performance in Open and Distance Learning, **2024. Published.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji	MDPI Electronics (Scopus-Indexed Journal. IF=2.6; CiteScore=5.3)	2079-9292	Chapters 3, 4 & 5
Ethical AI Framework for Integrating Artificial Intelligence in Open and Distance Learning, **2024. Accepted.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji, GE Jokthan, G Onwodi, KM Lawal, and CF Mafiana	8th EAI International Conference on Computer Science and Engineering. Held in Laredo, Texas, USA. (Scopus-Indexed Conference Proceedings Published by Springer, CiteScore=0.7)	1867-8211	Chapter 3
The Impact of Artificial Intelligence on Student's Academic Performance in Open and Distance Learning Using Multiple Regression Analysis Technique, **2024. Under Review.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji, GE Jokthan, G Onwodi, KM Lawal, and CF Mafiana	Journal of Infrastructure, Policy and Development (Scopus-Indexed Journal. CiteScore=1.0).	2076-3417	Chapters 3, 4 & 5
A Generalised Additive Modelling Approach to Uncovering the Influence of Artificial Intelligence on Student's Success Outcome in Distance Education, **2024. Under Review.**	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji, GE Jokthan, G Onwodi, KM Lawal, and CF Mafiana	AI (Scopus-Indexed Journal. IF=3.1; CiteScore=7.2).	2673-2688	Chapters 3, 4 & 5
Academic Performance Prediction in Distance Education: An Empirical Study Using Enhanced Support Vector Machine	MD Adewale, AA Azeta, A Abayomi-Alli, A Sambo-Magaji, GE Jokthan, G	AI (Scopus-Indexed Journal. IF=3.1; CiteScore=7.2).	2673-2688	Chapters 3, 4 & 5

Model, **2024. Under Review.**	Onwodi, KM Lawal, and CF Mafiana			

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (Contd.)

LIST OF TABLES

LIST OF ABBREVIATIONS

AAR	AI Alignment and Relevance
AI	Artificial Intelligence
AILA	AI-induced Learning Anxiety
API	Application Programming Interface
ARFC	AI Readiness and Facilitating Conditions
CAAI	Comparative Advantage of AI
CFI	Comparative Fit Index
EEU	Ease and Enjoyment of Use
GPA	Grade Point Average
IC	Interactive Capability

KAUS	Knowledge Absorption and User Satisfaction
MAE	Mean Absolute Error
MSE	Mean Squared Error
ODL	Open Distance Learning
RMSE	Root Mean Squared Error
SEM	Structural Equation Modelling
SQSI	Systems Quality and Social Influence
SVM	Support Vector Machine
TLI	Tucker-Lewis Index
VIF	Variance Inflation Factor

ABSTRACT

The integration of Artificial Intelligence (AI) in education, particularly within Open and Distance Learning (ODL) environments, presents substantial opportunities to enhance academic performance; however, a significant research gap exists in developing a comprehensive framework to predict the impact of AI adoption on student outcomes in ODL systems, especially considering moderating factors like gender and geographical context. This study aims to address this gap by designing, validating, and refining a predictive framework that leverages AI adoption factors to forecast academic performance in ODL settings. A predictive process framework was developed that leverages a mixed-methods approach by integrating Structural Equation Modelling (SEM) and Support Vector Machine (SVM) techniques. Data were collected from 914 students across diverse ODL environments through surveys, capturing variables related to AI adoption such as ease of use, knowledge absorption, and user satisfaction. The

SEM was utilized to validate the relationships between AI adoption factors and academic performance, achieving excellent fit indices (CFI and TLI = 1.000; RMSEA = 0.000). The SVM model was developed to predict academic performance based on the validated factors, with Variance Inflation Factor (VIF) optimization applied to address multicollinearity and enhance model stability. Model performance was evaluated using error metrics including Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). The results demonstrated that AI adoption significantly enhances academic performance when key factors are effectively integrated into ODL systems. The SEM confirmed strong relationships between AI adoption factors and academic performance, while the SVM model achieved high predictive accuracy (MAE = 0.229, MSE = 0.107, RMSE = 0.327). Although the improved SVM model showed a slight increase in error metrics (MAE = 0.295, MSE = 0.180, RMSE = 0.424), it provided more stable and reliable predictions. These findings indicate that the developed framework successfully predicts academic performance and underscores the importance of customizing AI tools to cater to diverse student needs, considering demographic variables such as gender and geographical location. In conclusion, by effectively integrating AI adoption factors into ODL systems, educators and policymakers can significantly enhance academic performance. The developed framework provides a practical tool for predicting and improving student outcomes, thus addressing the initial research gap and contributing to the advancement of AI in education. This implies that targeted AI integration can lead to better educational outcomes, especially when tailored to specific demographic contexts, highlighting the potential for ongoing improvements in AI applications within educational settings.

Keywords: Academic Performance, Artificial Intelligence, AI Adoption factors, Open and Distance Learning, Support Vector Machine.

CHAPTER ONE
INTRODUCTION

1.1 Background to the study

Over the years, the role of Artificial Intelligence (AI) in the educational sector has seen a considerable transformation. This evolution commenced in the 1960s with the advent of programmed instruction and the introduction of computer-assisted learning, establishing the foundational elements of AI within the educational realm. Moving into the 1980s, the field witnessed the rise of intelligent tutoring systems (ITS), which provided customized tutoring experiences by mimicking interactions in one-on-one tutoring situations. The proliferation of the internet during the 1990s and the early 2000s significantly propelled the use of AI in education, enabling the spread of online learning and the adoption of data-driven instructional strategies. The recent decade has been characterized by remarkable progress in machine learning and natural language processing technologies, leading to more advanced AI tools, including adaptive learning platforms and AI-enabled educational assistants. These innovations are pivotal in making education more personalized and widely accessible (Roll & Wylie, 2016).

Artificial intelligence (AI) has garnered significant popularity in educational settings, revolutionizing how people learn and teach. Educators leverage data-driven machine learning (ML) techniques and statistical frameworks to gain valuable insights into student performance patterns (Shen, Chen, Grey, & Su, 2021). AI technology is being utilized to enhance learning outcomes by creating digital labs, teaching platforms, and learning tools that cater for diverse learning needs (Lim, 2020). This approach provides students with personalized instructions, examples, and critiques and fosters the development of critical thinking skills (Wang, Liu, & Tu, 2021).

Contemporary research has started to measure the profound effects of AI on educational achievements. Studies have shown that AI-driven adaptive learning systems significantly enhance student engagement and academic performance. Specifically, adaptive learning systems have increased student participation by approximately 40% and improved test scores by an average of 30% compared to traditional teaching methods (Dabingaya, 2022). This underscores the effectiveness of AI in tailoring education to meet individual learner needs, thereby optimizing learning outcomes. Additionally, using AI in tutoring systems has been proven to reduce learners' study time by about 25% while maintaining or enhancing academic outcomes. This reduction highlights the efficiency of AI in delivering personalized educational experiences, which adapt to the unique learning styles and paces of students, making the educational process more effective (Gligorea, I., Cioca, M., Oancea, R., Gorski, A.-T., Gorski, H., & Tudorache, P., 2023). The application of AI extends beyond

enhancing student learning and performance to significantly reducing the workload for educators. Teachers who have implemented AI for administrative tasks have reported an average time saving of 5 hours per week. This additional time has allowed educators to focus more on direct student interaction and instructional improvement, illustrating the comprehensive benefits of AI in educational settings (Gligorea et al., 2023). These findings collectively highlight the transformative potential of AI in education, indicating improved efficiency, personalized learning experiences, and enhanced outcomes for both students and educators.

UNESCO (2019) highlights the role of AI in ensuring equal access to education for all individuals, including those with disabilities, refugees, and those in isolated communities. For instance, telepresence robotics enables students with special needs to attend school remotely, even in emergencies, ensuring inclusivity and accessibility across various locations. AI also revolutionizes collaborative learning by allowing students to choose when and where they study, regardless of their physical location. Asynchronous online discussion groups, a vital component of computer-supported collaborative learning, are monitored using AI systems like machine learning and shallow text processing. AI empowers teachers to gain insights into student discussions, guide their engagement, and enhance their learning experience.

Furthermore, AI facilitates personalized learning by supporting teachers in effectively assisting struggling students. With AI-powered dual-teacher models comprising a teacher and a virtual teaching assistant, routine tasks such as assigning homework and answering common questions can be streamlined. This enables teachers to allocate more time to individual student support and meaningful interactions, ultimately enhancing the quality of education. Many teachers are already embracing AI assistants to collaborate and optimize their teaching practices for the benefit of their students. AI plays a pivotal role in personalized learning and adaptive educational technologies, tailoring learning experiences to meet the diverse needs of students in Open and Distance Learning (ODL). Research has shown that AI-driven personalised learning systems can significantly impact student outcomes and engagement (Makokotlela, 2022). These technologies can adapt to individual learning styles and preferences, enhancing the learning experience.

Artificial Intelligence in Education (AIEd) has emerged as a significant focus in education, driven by advancements in communication and computing technologies (Chen et al., 2020; Hwang et al., 2020). The widespread use of AI tools in education has raised concerns and sparked discussions on improving student learning outcomes (Wang, Liu, & Tu, 2021). While AI in education is a globally recognized topic, its potential is not evenly realized across developed and developing countries.

Inclusion and equity in applying artificial intelligence in education pose significant challenges (Top 5 Challenges of Adopting AI in Education, 2021). Adopting AI in ODL presents challenges related to data privacy, algorithmic bias, and ethical considerations (Reis et al., 2020). Research has emphasized the impact of AI on the political landscape and the need to address data protection and ethical considerations in AI adoption (Reis et al., 2020). Challenges related to data privacy, algorithmic bias, and ethical considerations are critical in AI adoption in ODL. Issues such as data privacy violations and biased algorithmic decision-making can impact student outcomes and educational equity (Pillai & Sivathanu, 2020). Additionally, adopting AI in ODL may raise concerns about the ethical use of student data and the potential for algorithmic discrimination (Lee & Chen, 2022).

Despite the United Nations' efforts to enhance access to high-quality education and foster lifelong learning opportunities (UNESCO, 2022), gender differences persist in the motivation for advanced education involving AI technologies and applications (Squicciarini et al., 2020). However, there is limited evidence regarding the factors contributing to these gender differences, particularly in Africa. Hence, there is a critical need to conduct an in-depth examination of the factors influencing gender disparities in students' motivation to utilize AI technologies and applications within a pan-African framework. Previous systematic reviews have indicated that research on AI in education has primarily focused on developed countries (Roll & Wylie, 2016). As a result, AI in education remains a neglected topic in the developing world, where it is often considered part of an advanced technological discourse that relies on well-established infrastructure and knowledge ecosystems (UNESCO, 2019).

AI systems present new opportunities to promote gender equality and enhance the quality of life, potentially leading to increased productivity and improved job opportunities and services (European Commission, 2018). To foster gender inclusion and equality in adopting AI-based applications in education, researchers must broaden the scope of their studies and explore the factors contributing to gender differences in the utilization of these applications by students in higher education institutions within developing countries. By reducing barriers to learning access, automating administrative processes, and optimizing teaching methodologies to enhance student performance, AI holds great potential to accelerate the realization and development of global education goals (Padilla, 2019).

The utilisation of Artificial Intelligence (AI) technology has transformed various sectors, including education, where it has been increasingly utilised in ODL systems to enhance the teaching and learning processes (Chen et al., 2020; Shen et al., 2021). The acknowledgement of the potential for

AI adoption in ODL to improve students' academic performance through personalized learning experiences is widespread (Allam, Hassan, Mohideen, Ramlan, & Kamal, 2020). However, the precise impact of AI adoption on academic performance in ODL and how it differs based on factors such as gender and regional disparities between developing and advanced countries is still uncertain. Therefore, further research is required to investigate this matter. Some studies have focused on exploring factors influencing student persistence in ODL, identifying both success factors and challenges faced by students in this mode of learning and proposing strategies for enhancing student persistence based on their findings (Au, Li, & Wong, 2018), the direct impact of AI adoption on academic performance remains largely unexplored.

Assessment in ODL not only serves as a means of grading and certifying students but also plays a critical role in their learning improvement and monitoring the effectiveness of academic programs, enabling the adoption of appropriate strategies to achieve institutional objectives (Koneru, 2017). In a recent study conducted during the COVID-19 pandemic, Libasin et al. (2021) compared the influence of different learning styles on students' academic performance between synchronous and asynchronous online learning in a Malaysian university. The findings revealed a positive impact of synchronous online learning on students' academic performance compared to asynchronous online learning. However, it is important to recognize the impact of AI adoption on academic performance. The variability of AI is contingent upon the precise context and manner of its implementation within ODL (Shen et al., 2021). Therefore, further research is necessary to delve into the effects of AI adoption on academic performance in ODL, considering the potential differences that may arise based on gender and regional disparities.

ODL is being increasingly adopted to expand access to education and enhance the development of digital skills, leveraging the opportunities presented by digital technologies. Within this context, Artificial Intelligence (AI) is an emerging field with numerous applications, including education. Machine learning algorithms, natural language processing, and computer vision are widely used in AI adoption in ODL to analyze student data, provide personalized learning experiences, and optimize educational content delivery (Valentin et al., 2022; Mathew & Chung, 2021). These methodologies have been instrumental in understanding student perceptions and enhancing the implementation of ODL amidst the COVID-19 pandemic (Mathew & Chung, 2021). For example, Machine learning algorithms have been widely used to predict student academic performance based on historical data and learning patterns. Various studies have demonstrated the effectiveness of machine learning techniques in this context. For instance, Yağcı (2022) compared the performances of different machine learning algorithms such as random forests, nearest neighbour, support vector machines,

logistic regression, Naïve Bayes, and k-nearest neighbour to predict students' final exam grades.

Similarly, Livieris et al. (2018) applied supervised learning algorithms to develop accurate models for predicting student characteristics that influence their behaviour and performance. Onyema et al. (2022) and Buenaño-Fernández et al. (2019) also utilized machine learning algorithms to forecast students' academic outputs and predict the final grades of students based on their historical performance, respectively. Predicting academic performance through machine learning algorithms, particularly support vector machines (SVMs), is a notable area of AI research in education, specifically in ODL environments. This is the first study to look into the impact of AI adoption on academic performance in ODL settings using SVM. Furthermore, this study adds to the small body of literature addressing this current and critical issue related to Africa.

AI within ODL environments offers educators and institutions a comprehensive opportunity to enhance educational delivery. This integration necessitates a significant transformation in curriculum design and teaching methodologies, highlighting the critical importance of AI tools in contemporary educational scenarios. Togaibayeva et al. (2022) discuss the transformative potential of embedding AI technologies within educational frameworks, showcasing the broad possibilities for innovation. Concurrently, Sakibayev et al. (2019) provide evidence of the academic advantages stemming from the application of mobile technology in database courses, demonstrating clear, positive impacts on student achievement and success.

The journey toward adopting AI in education is complex, requiring a holistic view that encompasses a range of considerations—from technological advancements to socio-political, economic, cultural, and ethical dimensions. This comprehensive approach is supported by the work of Namoun & Alshanqiti (2020), Tait & Godfrey (2001), Shen (2023), Oyedeji et al. (2020), and Babić (2017), whose research collectively deepens our understanding of AI's advantages and limitations within educational contexts. As AI in education evolves rapidly, its capacity to fundamentally transform teaching and learning practices becomes increasingly evident. The development of predictive models, such as those utilizing Support Vector Machines (SVMs), to gauge the impact of AI adoption on student academic performance in ODL exemplifies just one avenue through which AI can significantly enhance educational outcomes. This innovative approach not only aligns with the broader trajectory of educational technology but also serves as a testament to the potential of AI to facilitate a more adaptive, personalized learning experience for students across diverse learning environments.

1.2 Statement of the problem

Despite the rapid adoption of AI in ODL, there is a critical gap in understanding how AI impacts academic performance. Existing research lacks a comprehensive framework for predicting these effects, leaving educators and institutions uncertain about how to best leverage AI to improve learning outcomes (García-Martínez et al., 2023; Alonso et al., 2021). This gap hinders the ability to make data-driven decisions that optimize AI's benefits in educational settings.

A key issue is the absence of studies that address the moderating role of gender and the contextual differences between developed and developing countries. These factors significantly shape how students interact with AI technologies, yet their impact on academic performance remains underexplored. For instance, gender may influence technology adoption and learning engagement, while students in developing countries face unique challenges such as limited access to AI tools (UNESCO, 2022; Yannier et al., 2021). This research tackles these gaps by developing a predictive framework using Support Vector Machine (SVM) to assess the impact of AI on academic performance in ODL systems. Crucially, it investigates how gender and contextual factors in both developed and developing countries affect this relationship. By addressing these nuances, this study provides actionable insights to optimize AI use in diverse educational contexts, making it a timely and necessary contribution to the field.

.

1.2.1 Research Questions

The following are the research questions for the study:

I. What are the requirements for adopting AI in Open Distance Learning (ODL) (Dua, 2021)?
II. How can a process model that incorporates AI requirements in ODL be designed?
III. How can a research model be designed to incorporate factors of AI and student academic performance?
IV. How can machine learning models be developed with impact factors of AI adoption and student academic performance (Namoun & Alshanqiti, 2020)?
V. How can machine learning models of AI adoption and student academic performance be evaluated to determine the level of accuracy (Valentin et al., 2022)?

1.2.2 Research Hypothesis

The following are the research hypotheses for the study:

I. Comprehensive requirements of AI adoption in ODL would enhance student academic performance.

II. The design of a process framework would enhance the understanding of AI adoption in ODL.
III. The factors of AI adoption have a significant impact on student academic performance.
IV. The developed machine learning models would predict the impact of AI adoption on ODL students' academic performance.
V. The evaluation of the machine learning models would have a significant impact on the model's accuracy.

1.3 Aim of the study

This research aims to develop a process framework for predicting the impact of artificial intelligence adoption on students' academic performance in Open and Distance Learning (ODL) using a support vector machine.

1.4 Specific objectives

The specific objectives are to:

I. To develop a process framework incorporating the factors identified from the requirements to enhance understanding of AI adoption in ODL.
II. To develop a research model comprising the factors of AI adoption and student academic performance in ODL.
III. To develop a machine learning model to predict the impact of the identified factors of AI adoption on student academic performance.
IV. To evaluate the machine learning models to establish the level of accuracy.

1.5 Scope of the Study

The scope of the study includes:

I. **Target population:** The research focuses on students in Open and Distance Learning (ODL) systems currently enrolled in courses using AI-based interventions in Canada and Nigeria.
II. **Variables:** The study examines the impact of AI adoption on students' academic performance in ODL systems, focusing on factors such as student engagement, course design, and the effectiveness of AI-based interventions.
III. **Methodology:** The research designs a process-based framework and implements the framework using a Support Vector Machine (SVM) algorithm to predict AI adoption's impact on academic performance in ODL systems. The research methodology strongly emphasises assessing Moodle's AI capabilities, given its prominence and comprehensive utilization in ODL settings.

Moodle's AI tools are grounded in literature as the most evaluated AI solutions for fostering AI adoption in educational contexts, making them an essential focus for this research.

IV. **Data Collection:** Data are systematically gathered via surveys distributed to a sample of students in ODL systems, with questions tailored to gauge the utility and impact of Moodle's AI tools on their learning outcomes.

1.6 Significance of the Study

This research makes a significant contribution by developing a predictive framework that Open and Distance Learning (ODL) institutions can adopt to assess the impact of AI integration on students' academic performance. It addresses a critical gap in the existing literature by providing a comprehensive tool for ODL stakeholders, enabling them to evaluate both the benefits and potential drawbacks of AI adoption in their unique contexts. The framework is designed to predict academic outcomes by incorporating key AI adoption factors, as well as moderating influences such as gender and geographical differences, which have been underexplored in prior studies.

By applying a Support Vector Machine (SVM) model, this study provides a novel approach to forecasting academic performance in ODL settings. The model accounts for complex interactions between AI adoption factors and performance, offering insights that can inform policy decisions and educational strategies. Additionally, this framework can be applied across different educational disciplines, broadening its applicability beyond ODL environments.

The study's practical significance lies in its potential to assist educators and policymakers in making data-driven decisions about AI's role in improving academic outcomes. The theoretical contributions include advancing our understanding of AI's impact on learning environments, particularly in developing countries, where infrastructural constraints play a significant role. Methodologically, the research introduces an innovative process-based framework combined with predictive analytics, creating a scalable and replicable tool for evaluating AI's effects in education.:

1.7 Definition of Terms

In the context of this research, it is essential to clarify and define specific terms to ensure a unified understanding and to avoid ambiguities. Here are the definitions for the critical terms used throughout the thesis:

I. **Artificial Intelligence (AI):** Refers to the simulation of human intelligence processes by machines, especially computer systems. These processes include learning, reasoning, and self-correction.

II. **Open and Distance Learning (ODL):** A mode of education that caters to learners who might not be physically present in traditional classroom settings. It often leverages technology to deliver content and facilitate communication.

III. **Support Vector Machine (SVM):** A supervised machine learning algorithm that can be employed for both classification or regression tasks. It functions by finding a hyperplane in an N-dimensional space that distinctly classifies the data points.

IV. **Unified Modeling Language (UML):** A standardized modelling language can visualize a system's architectural blueprints, including activities, actors, business processes, and system components.

V. **Dataset:** A collection of data, typically organized in tabular form, where each row represents an instance and columns represent the attributes of the instance.

VI. **Validation:** The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies the specified requirements.

VII. **Research Methodology:** A systematic way to solve a problem. It is the science of studying how research is conducted scientifically.

VIII. **Literature Review:** An evaluation of existing research related to the topic. It helps identify gaps, contradictions, parallels, and complements in the literature.

IX. **Performance Metrics:** Quantitative measures used to assess the performance of algorithms or models.

Each term, as defined above, serves as a foundation for the discussions and analyses that follow in the thesis. Understanding these terminologies aids in comprehending and appreciating the research's depth and implications.

1.8 Organization of the Thesis

The thesis has been meticulously organized to provide a comprehensive and coherent understanding of the study's progression. Chapter 1, titled "Introduction," offers a contextual backdrop, setting the stage for the research by defining the problem, outlining the overarching intent, specific goals, boundaries, and significance, while providing essential terminologies for clarity. Chapter 2 reviews previous studies on AI adoption in ODL and gives literature on subjects that include the impact of AI on academic performance, AI-driven personalized learning, and the use of Support Vector Machines (SVM) for predictive modeling in education. It also discusses theoretical frameworks and provides a consolidated meta-analysis to identify gaps and patterns from reviewed studies. Chapter 3 describes the empirical study carried out. This includes the development of a process framework for predicting academic performance, data collection from ODL students, model implementation using SVM and Structural Equation Modeling (SEM), and the evaluation of moderating factors such as gender and

geographical context. The chapter also elaborates on the use of Unified Modeling Language (UML) to visualize the system's architecture and discusses performance metrics used to validate the model. Chapter 4 presents and discusses the implications and significance of results obtained, focusing on the predictive accuracy of AI models in assessing academic performance in ODL settings and their potential for optimizing educational strategies. The results are supported by quantitative measures, visual aids such as charts and graphs, and comparisons of the AI models' outcomes with traditional educational technologies to assess their effectiveness. Finally, Chapter 5 summarizes the study and concludes with suggestions for future research, including refining the models, extending the framework to other educational environments, and offering practical advice for educators, policymakers, and technologists on optimizing AI deployment in learning environments to foster more equitable, efficient, and engaging educational experiences.

1.9 Limitations of the Study

The limitations of this study are important to consider when interpreting the findings. First, the generalizability of the results may be limited, as the study focuses on a specific population of students in ODL systems, which may not fully represent all ODL contexts. Additionally, the sample size, while comprehensive enough for the analysis conducted, may still limit the broader applicability of the findings, particularly in diverse educational settings or regions. Another limitation is the reliance on self-reported data from students, which introduces the potential for social desirability bias. This means that students may have responded in ways they perceived as favorable rather than completely reflecting their true experiences or opinions. Despite these limitations, the study offers valuable insights into the relationship between AI adoption and academic performance in ODL environments, and it serves as a foundation for future research in this area. Future studies could address these limitations by expanding the sample size, including diverse populations, and utilizing more objective data collection methods.

.

CHAPTER TWO
LITERATURE REVIEW

2.1 Preamble

Integrating AI in education has unveiled unprecedented opportunities for the enhancement of students' academic achievements. Specifically, ODL has experienced a notable increase in the application of AI to optimize both the learning experience and educational outcomes. Assessing the ramifications of AI integration on students' academic performance within ODL is of paramount significance for educational institutions, policymakers, and scholars alike. The objective of this chapter is to present the theoretical framework, a methodical review of pertinent literature, and an examination of related scholarly works to investigate and evaluate existing research concerning the specified subject matter. Through the adoption of a systematic and exhaustive methodology in the literature review, this analysis aims to discern the principal factors affecting AI adoption, investigate the predictive efficacy of SVM in evaluating the influence of these principal factors on students' academic performance, and reveal the impact of moderating variables such as gender and regional differences that may affect this influence.

Through a comprehensive exploration of esteemed databases, including Web of Science, Scopus, Google Scholar, and an array of pertinent articles published from 2015 to 2023, a systematic review of relevant literature and an examination of related works were conducted. The chosen articles underwent a meticulous assessment, emphasizing their coherence with the research inquiries and the rigour of the research methodologies utilized. Only peer-reviewed articles authored in English were deemed acceptable to ensure the dependability and trustworthiness of the findings. By integrating the insights derived from the selected articles, this systematic review of pertinent literature and the examination of related works aspires to furnish insightful perspectives regarding the procedural framework for forecasting the influence of AI adoption on students' academic performance in ODL employing SVM. The results of this review have the potential to enhance the integration of AI within ODL environments, guide decision-making processes for educational institutions, and facilitate further research within this swiftly advancing domain.

This systematic literature review is anticipated to shed light on the existing knowledge gaps, offer recommendations for future research, and provide a comprehensive understanding of the factors influencing the impact of AI adoption on students' academic performance in ODL. Ultimately, this review aims to contribute to the ongoing discourse on leveraging AI technologies to optimize the educational experience and outcomes for distance learners. This chapter addresses the following:

- **Theoretical Framework**
  - Technology Acceptance and Adoption Models
  - Support Vector Machine (SVM) approach
  - Conceptual framework of the study
  - Theoretical assumptions of the study
- **Review of Relevant Literature**
  - Factors Driving AI Adoption in ODL
  - Impact of AI Adoption on Academic Performance in ODL
  - Predicting Academic Performance Using SVM
  - Moderating Factors: Gender and Regional Differences
- **Review of Related Works**
- **Summary/Meta-Analysis of Reviewed Related Works**

2.2 Theoretical Framework

The exponential progression in artificial intelligence (AI) and its integration into education has generated significant interest among researchers. One crucial aspect is the impact of AI adoption on students' academic performance, particularly in the context of Online Distance Learning (ODL). This study aims to develop a process framework utilizing a Support Vector Machine (SVM) to predict the impact of AI adoption on students' academic performance in ODL. The theoretical framework is the cornerstone of any research, laying the foundation for interpreting the dynamics and outcomes of the study. In the context of this present work, the theoretical framework is instrumental in guiding the exploration and analysis of critical components such as AI adoption factors, moderating factors, and the outcome variable of students' academic performance.

This study integrates various theories and models to investigate the impact and adoption of AI in Open and Distance Learning (ODL) environments. The Technology Acceptance and Adoption Models primarily utilized include the TAM or the Technology Acceptance Model, UTAUT or the Unified Theory of Acceptance and Use of Technology, and the D&M Model or Information Systems Success. These models offer valuable insights into students' acceptance and utilization of AI in the ODL framework. They highlight the significant roles played by factors such as social influence, perceived ease of use, and perceived usefulness in determining the effective use and adoption of AI technology.

The Information Systems Success (D&M Model) is a foundation for understanding the factors contributing to AI adoption's success in ODL. It focuses on system quality, information quality, service quality, and user satisfaction, offering a comprehensive perspective on the effectiveness of integrating AI in ODL settings. By examining these theories, a comprehensive understanding of the complex dynamics involved in AI adoption in ODL and its impact on student performance can be gained. Applying the Support Vector Machine (SVM) algorithm aims to predict and analyse the relationships between AI adoption, ODL environments, and student outcomes. This predictive capability allows us to assess the potential of AI in enhancing educational experiences and outcomes in ODL. Utilizing these theories and models aims to provide valuable insights for educators, institutions, and policymakers seeking to leverage AI in ODL effectively. This research contributes to the broader understanding of how AI adoption can positively influence student performance and establish a more effective and personalized learning environment in the ODL landscape.

Finally, the principles underlying the SVM algorithm provide the technical underpinning for constructing the predictive model. This tool forecasts academic performance based on AI adoption and associated factors. The theoretical framework of this study intertwines a series of complex theories and models to form a coherent, insightful, and effective tool for predicting the impact of AI adoption on students' academic performance in ODL using SVM.

2.2.1 Technology Acceptance and Adoption Models

For an organization to successfully adopt modern technologies, it is important to understand the factors influencing their adoption thoroughly. This knowledge is crucial for effective planning and implementation, as it allows the organization to minimize internal and external pressures. By adopting modern technologies in a planned and strategic manner, organizations can ensure a smoother transition and maximize the benefits gained from using these technologies (Javaid et al., 2022; Birajdar & Vasudevan, 2022). Therefore, the primary goal of this review is to review different technology acceptance methods and identify the most effective technology acceptance method or combination of methods that are used in evaluating the factors that influence AI adoption in ODL settings. This section focuses on the theoretical frameworks that aim to understand and explain individuals' acceptance and adoption of new technologies. It discusses the models that provide valuable insights into the factors influencing individuals' attitudes and intentions towards adopting and using a particular technology. By examining users' perceptions, beliefs, and behaviours, TAAMs help researchers and practitioners understand how technology can be effectively introduced and integrated into various domains, including education. These Technology Acceptance and Adoption Models provide valuable frameworks for understanding and predicting users' acceptance and

adoption of technologies in various contexts, including education. By considering factors such as perceived usefulness, ease of use, social influence, and individual beliefs, these models assist in designing strategies and interventions to promote the successful adoption and implementation of educational technologies, such as artificial intelligence, in educational settings.

Technology acceptance model (TAM)

The technology acceptance model is one of the most important models for how people accept new technology. An individual's intention to use new technology is primarily shaped by the perceived ease of use (PEOU) and the perceived usefulness of the technology (PU). The likelihood of an older adult learning digital games depends on their perception of the games. If they believe learning how to use digital games will be too difficult or a waste of time, they will be less likely to adopt this technology. However, if they believe digital learning games will provide much-needed mental stimulation and be easy to understand, they will be more likely to want to learn how to use digital games. While TAM has been criticized frequently, it remains a practical general framework consistent with several studies into the factors influencing older adults' willingness to use new technology (Charness & Boot, 2016). This model's emphasis on the potential user's perceptions is crucial. Figure 2.1 depicts TAM's original theoretical framework.

The model shows that behavioural intention (BI) determines the actual use of the system (AU), and BI is jointly and directly determined by one's attitude toward using the system (ATT) and one's PU. PU and PEOU have an impact on attitude as well. The inquiry centred on the utilization of the extended Technology Acceptance Model (TAM) to investigate how four factors influence Home Economics (HE) teachers' behavioural intention (BI) to use the Internet as a teaching tool, namely Internet attitude (IA), perceived ease of use (PEOU), and perceived enjoyment (PE), perceived usefulness (PU). The findings indicated that HE teachers' BI positively correlates with IA, PU, PEU, and PE (Phua, Wong, & Abu, 2012). This study employs the constructs of the proposed research model (PEOU, PU, BI, ATT, and PE).

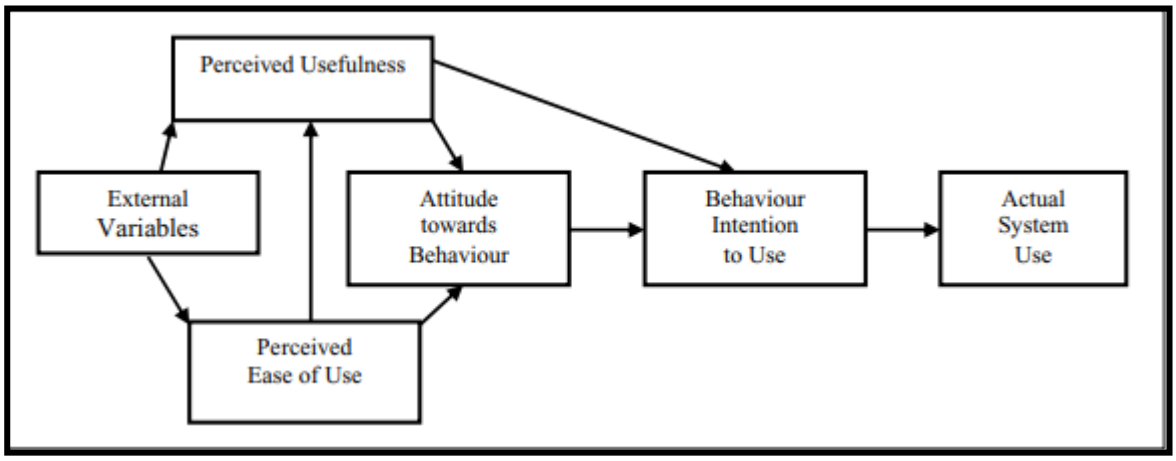


Figure 2.1 Original Technology Acceptance Model (TAM) Source: (Phua, Wong, & Abu, 2012)

Information Systems Success (D&M Model)

Intelligent systems cost much money and take time and work to set up. So, researchers and practitioners are always trying to figure out the most important factors that affect how these systems are used and how successful they are. According to (Sabeh et al., 2021), the DeLone and McLean (D&M) success model is one of the most common ways to study technology success. Many scholars have added to and improved the original D&M model or pointed out problems. The DeLone and McLean (D&M) information systems (IS) success model aims to give a complete picture of IS success by figuring out and explaining the relationships between the most critical success factors. According to (Ojo, 2017), the model provides six interconnected dimensions of IS success.

These are system quality, information quality, service quality, (intentional) use, user satisfaction, and net benefits. Numerous IS studies have used the D&M model and confirmed its validity. For example, Hospital information systems in developing countries were the focus of the research work by Ojo, 2017, which is an adaptation of the well-known DeLone and McLean information system success model. It was found that the quality of the system and the frequency with which it is used are significant indicators of a thriving hospital information system. This present study considers system quality and user satisfaction alongside other constructs from other technology acceptance models.

The UTAUT, or Unified Theory of Acceptance and Use of Technology

The UTAUT has gained significant attention from scholars in the technology acceptance field, according to Yakubu and Dasuki (2018). This is attributed to UTAUT's holistic framework facilitating a nuanced comprehension of the factors influencing technology adoption and usage. Consequently, UTAUT has become a widely used theoretical framework for research on technology adoption. The hedonic motivation, price value, and habit are the three additional constructs taken into consideration by the UTAUT2 model, which is an updated version of the framework used initially for the UTAUT model. The original UTAUT model contained four variables: facilitating conditions,

performance expectation, effort expectation, and social influence (SI). Yakubu and Dasuki (2018) researched higher education students in Nigeria based on the UTAUT model. They found that the promotion conditions and behavioural intention were critical factors affecting their use of educational technology. Ameri et al. (2020) used a modified UTAUT2 questionnaire to survey pharmaceutical students. The results showed that social influence (SI) and performance expectancy (PE) positively affected behavioural intention. Almaiah et al. (2019) used the UTAUT model to explain why higher education students accepted a mobile learning system. They found that the main reasons were perceived information quality and perceived security. This present study considers facilitating conditions, social influence (SI), and other constructs from other technology acceptance models.

2.2.2 Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a powerful machine learning algorithm that has gained popularity and success in various domains, including classification and regression tasks. SVM is based on the theoretical foundations of statistical learning theory and optimization techniques (Yang et al., 2023). This theoretical framework provides a solid basis for understanding the principles and concepts underlying the SVM approach. Here, the theoretical foundations of SVM as a machine learning algorithm are explored. The principles and mathematical concepts underlying SVM's ability to predict and classify data will be examined in relation to its application in predicting the impact of AI adoption on students' academic performance in ODL.

I. Statistical Learning Theory

Statistical Learning Theory forms the basis of the SVM approach. It focuses on the analysis of data to make predictions or decisions. The key idea behind statistical learning theory is to find a function that can accurately generalize from observed data to unseen instances. SVM leverages statistical learning theory to construct a decision boundary that maximizes the margin between different classes, aiming to achieve better generalization performance (Yang et al., 2023).

II. Linear Separability and Kernel Trick

SVM is based on the assumption that the data points of different classes can be separated by a hyperplane in a high-dimensional feature space. This assumption is known as linear separability. However, the data may not be linearly separable in the original feature space. The kernel trick transforms the data into a higher-dimensional space where linear separability can be achieved. The choice of appropriate kernel functions, such as linear, polynomial, or radial basis function (RBF), plays a crucial role in SVM's performance (Ramus et al., 2023; Singam et al., 2023; Rukhsar et al., 2022).

III. Margin Maximization and Support Vectors

The SVM algorithm seeks to identify the hyperplane that optimizes the margin separating the support vectors, which represent the data points nearest to the decision boundary. The margin represents the separation between different classes and provides a measure of robustness against noise and outliers. By maximizing the margin, SVM promotes better generalization and improved classification accuracy (Petrova & Bojikova, 2022; Rizwan et al., 2021).

IV. Convex Optimization

SVM involves solving a convex optimization problem to find the optimal hyperplane. Convex optimization techniques, such as quadratic programming, are utilized to determine the hyperplane parameters that minimize the classification error and maximize the margin. The convexity of the optimization problem guarantees the solution's global optimality and ensures the SVM algorithm's efficiency (Wang et al., 2021; Piccialli & Sciandrone, 2022).

V. Regularization and Control of Overfitting

Overfitting is a common issue in modelling where a model may perform well on the training data but fails to generalize to novel, unseen data. To address this issue, Support Vector Machines (SVMs) utilize regularization techniques such as the C parameter, which controls the balance between achieving a larger margin and minimizing the classification error. Regularization helps prevent overfitting by introducing a penalty for misclassified instances and balancing the complexity of the model (Ghojogh & Crowley, 2019; An et al., 2020).

The theoretical framework of the Support Vector Machine (SVM) approach is grounded in statistical learning theory, optimization techniques, and convex optimization. By leveraging the concepts of linear separability, margin maximization, support vectors, and convex optimization, SVM provides a robust and efficient method for classification and regression tasks (Chopra & Khurana, 2023; Sun, 2016). Understanding the theoretical foundations of SVM is essential for effectively utilizing and interpreting the results of this powerful machine-learning algorithm. The principles and mathematical concepts underlying SVM contribute to its predictive ability. SVM's performance in various domains, such as breast cancer prediction, compressor performance prediction, and academic performance prediction, indicates its effectiveness as a machine learning algorithm based on statistical learning theory and optimization techniques (Yang et al., 2023).

2.2.3 Conceptual Framework

The conceptual framework outlines the fundamental components and interrelations in forecasting the influence of AI adoption on the academic performance of students. This framework encompasses three essential dimensions: Factors Influencing AI Adoption, Moderating Variables including Gender and Regional/Geographical Disparities, and the Resultant Variable.

I. Factors Influencing AI Adoption

The factors influencing AI adoption within the realm of Online Distance Learning (ODL) encompass a diverse array of elements that significantly affect the integration and utilization of AI technologies. These determinants illuminate the complex interactions between AI systems and essential stakeholders, including educational institutions, instructors, and learners. As posited by Chen et al. (2020), AI platforms must be congruent with the objectives, ethical standards, and requirements of the learning community, providing unique benefits compared to conventional educational methodologies, such as tailored learning experiences and improved engagement. Simultaneously, factors related to AI preparedness evaluate an institution's technical capabilities, cognitive readiness, and the existing infrastructure for AI integration. Moreover, an institution's flexibility, in conjunction with factors such as technology accessibility, motivation, and perceived usefulness, plays a crucial role in shaping the landscape of AI adoption. Such elements can be systematically assessed through surveys, facilitating a thorough quantitative investigation of AI adoption (Phua, Wong, & Abu, 2012).

II. Moderating Factors, Including Gender and Regional Variations

The moderating factors denote those elements that influence the effectiveness of AI adoption factors on students' academic performance in ODL. The impact of AI adoption factors is moderated by variables such as gender and regional/geographical variations, thereby affecting the relationship between AI adoption and academic performance. Acknowledging these moderating factors is essential as they provide critical insights into how diverse demographics may react differently to AI adoption in ODL, ensuring a comprehensive understanding of the dynamics at play.

III. Dependent Variable

The dependent variable in this study pertains to the academic performance of students, which can be assessed through various indicators such as final grades, assessment scores, or cumulative GPA. This academic performance functions as the dependent variable, which is forecasted based on the factors of AI adoption and the moderating elements.

IV. Support Vector Machine (SVM)

The support vector machine (SVM) is an algorithm in machine learning that is amenable to deploying within the framework of predicting students' academic performance by identifying artificial intelligence (AI) adoption factors and moderating factors. SVM has shown promising results in predicting students' achievements, engagement, and performance in online learning settings, and it achieves this by utilizing a classification or regression approach to construct a predictive model. This model can categorize students into different performance classifications or estimate their performance levels (Tomasevic et al., 2020; Ayouni et al., 2021).

The conceptual framework (Refer to Figure 2.2) asserts that factors influencing the adoption of AI have a significant impact on students' engagement with AI in ODL, thereby subsequently affecting their academic outcomes. The moderating variables act as intermediary constructs, facilitating the prediction and clarification of the correlation between AI adoption and academic performance. The SVM algorithm develops a predictive model aimed at forecasting students' academic success based on the identified factors and moderating elements.

By taking into account the factors related to AI adoption, the moderating variables, and harnessing the capabilities of the SVM algorithm, the proposed conceptual framework outlines a systematic approach to investigate the ramifications of AI adoption on students' academic performance within the ODL environment. This framework aspires to enhance our comprehension of the interrelationship between AI adoption and academic results in the context of ODL. The insights obtained from this inquiry may prove instrumental in devising effective strategies for the integration of AI in educational settings and in optimizing students' learning experiences within online platforms.

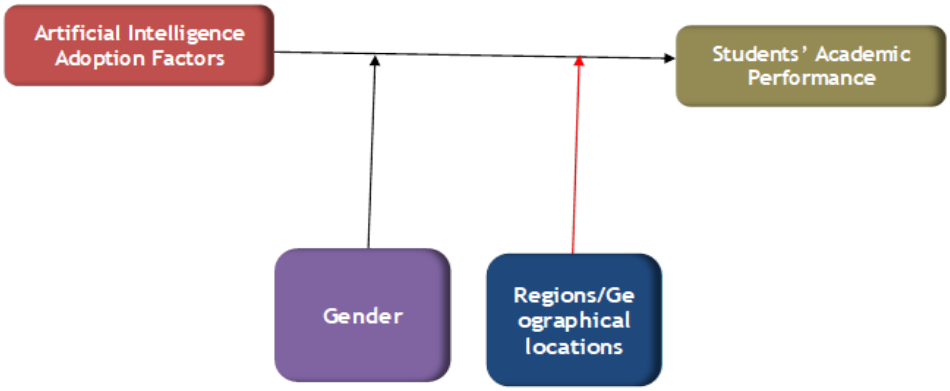


Figure 2.2 Conceptual Framework (Source: Author's work)

2.2.4 The Theoretical Assumptions of the Study

The theoretical assumptions of the study include the following:

- **Technological Determinism:** This study assumes that adopting AI technology can directly and

significantly impact students' academic performance in an ODL (Open and Distance Learning) environment.

- **Adoption Factors Matter:** The research posits that certain adoption factors (e.g., acceptance, access, motivation, perceived usefulness, and ease of use) play a critical role in the successful integration and utilization of AI in education.
- **The Moderation Effect:** The study presumes that gender and regional/geographical differences can moderate the relationship between AI adoption and students' academic performance.
- **Measurability:** The research assumes that both the AI adoption factors and students' academic performance can be accurately measured using available tools and techniques (e.g., validated scales, surveys, and grade point averages).
- **Predictability:** A key assumption is that the student's academic performance can be predicted using SVM, a machine learning algorithm based on AI adoption factors and moderating factors.
- **Universal Applicability of AI:** The study assumes that AI technologies can be effectively used in various educational contexts and disciplines within ODL.
- **Technological Readiness:** It assumes that the technology infrastructure in the ODL environment is ready to accommodate the use of AI technologies.
- **Causal Relationships:** The research assumes that the relationships between AI adoption factors, moderating factors, and academic performance are causal, not merely correlational.
- **Transferability:** The study assumes that the results and findings are generalizable and can be transferred to other similar educational contexts.
- **Technological Neutrality:** The study assumes that AI is a neutral tool, the effects of which are determined by how it is used in the ODL environment rather than the inherent qualities of the technology itself.

These assumptions form the theoretical backbone of the study, guiding its design, execution, and interpretation of results. It is important to remember that these assumptions would need to be scrutinized and tested as the study progresses to ensure the validity and reliability of the findings.

2.3 Review of Relevant Literature

AI offers numerous opportunities for Open and Distance Learning (ODL) institutions, specifically in addressing effective teaching and learning methods and exploring the advantages and limitations of computer-based systems in education (Liu & Huang, 2022). The flexibility and accessibility of ODL have encouraged more female students to study IT and computer science (Ogunsola-Bandele & Kennepohl, 2022). Integrating AI into distance education can profoundly impact instructional methods, guidance approaches, and educational content (Gao, 2022). By incorporating cutting-edge

AI technology into existing e-learning systems, personalized, adaptive, and intelligent services can be provided to students and educators alike (Tanjga, 2023). However, the full implementation of AI in education has not been fully realized, and successful AI applications in e-learning have yet to be widely adopted, especially in open-source learning management systems (Huang et al., 2021).

Kuleto et al. (2021) state that Artificial Intelligence (AI) and Machine Learning (ML) have their roots in data management and development processes. Integrating AI and ML into various industries, including education, is a groundbreaking trend. This integration enhances learning by customizing platforms and applications to meet student needs. Extensive research is underway to improve educational processes, making AI in Education a rapidly advancing field within the education sector. Developed countries have displayed significant interest in exploring the applications of AI in Higher Education, leading to a wealth of literature on this subject. AI and ML technologies enhance education by fostering student competence, facilitating group work, and providing easy access to academic resources. With the increasing prominence of AI tools, there is a growing emphasis on their utilization in educational settings to enhance students' learning performance.

In recent years, the adoption of AI has become widespread across various industries, including education. Integrating AI into Open and Distance Learning (ODL) has the potential to enhance student's learning experiences and improve their academic performance. However, there is a need to develop a process framework that can predict the impact of AI adoption on students' academic performance in ODL. This systematic literature review aims to identify relevant studies, synthesize their findings, and propose a process framework for predicting the impact of AI adoption on students' academic performance in ODL.

The systematic literature review followed a well-established methodology involving the identification of relevant studies, data extraction, and synthesis of findings (de la Torre-López, Ramírez, & Romero, 2023). The review focuses on several aspects of AI adoption in ODL, including the factors driving its adoption, the impact of AI adoption on students' academic performance, the use of a Support Vector Machine (SVM) for predicting this impact, and potential gender and regional differences in the effect of AI adoption on academic performance in ODL. The review followed a four-step process, as depicted in Figure 2.3:

I. Identification of relevant studies,
II. Screening of studies,
III. Eligibility/selection of studies, and
IV. Inclusion of studies and synthesis of findings.

A systematic search was performed using Google Scholar, Scopus, and Web of Science databases, as well as a snowballing approach, to conduct this review. A total of 700 studies were identified, of which 80 were selected for full-text screening. After the screening, 53 studies were included in the final selection. The studies were published between 2015 and 2023 and were conducted in different countries, including the United States, China, and India. The search terms used included "artificial intelligence adoption," "online distance learning," "academic performance," "support vector machine," "gender differences," and "regional differences." The inclusion criteria encompassed peer-reviewed articles published between 2015 and 2023, written in English, and directly relevant to the research questions. Articles not peer-reviewed, unrelated to the research questions, or published before 2015 were excluded. After an extensive literature search, 53 articles addressing the research questions were identified. The following research questions guide the systematic literature review:

I. What factors drive AI adoption in Online Distance Learning (ODL) settings?
II. How does AI adoption impact students' academic performance in ODL?
III. How can these factors be used to predict students' academic performance using the Support Vector Machine (SVM) approach?
IV. How do moderating factors such as gender and regional differences affect the impact of AI adoption on students' academic performance in ODL?

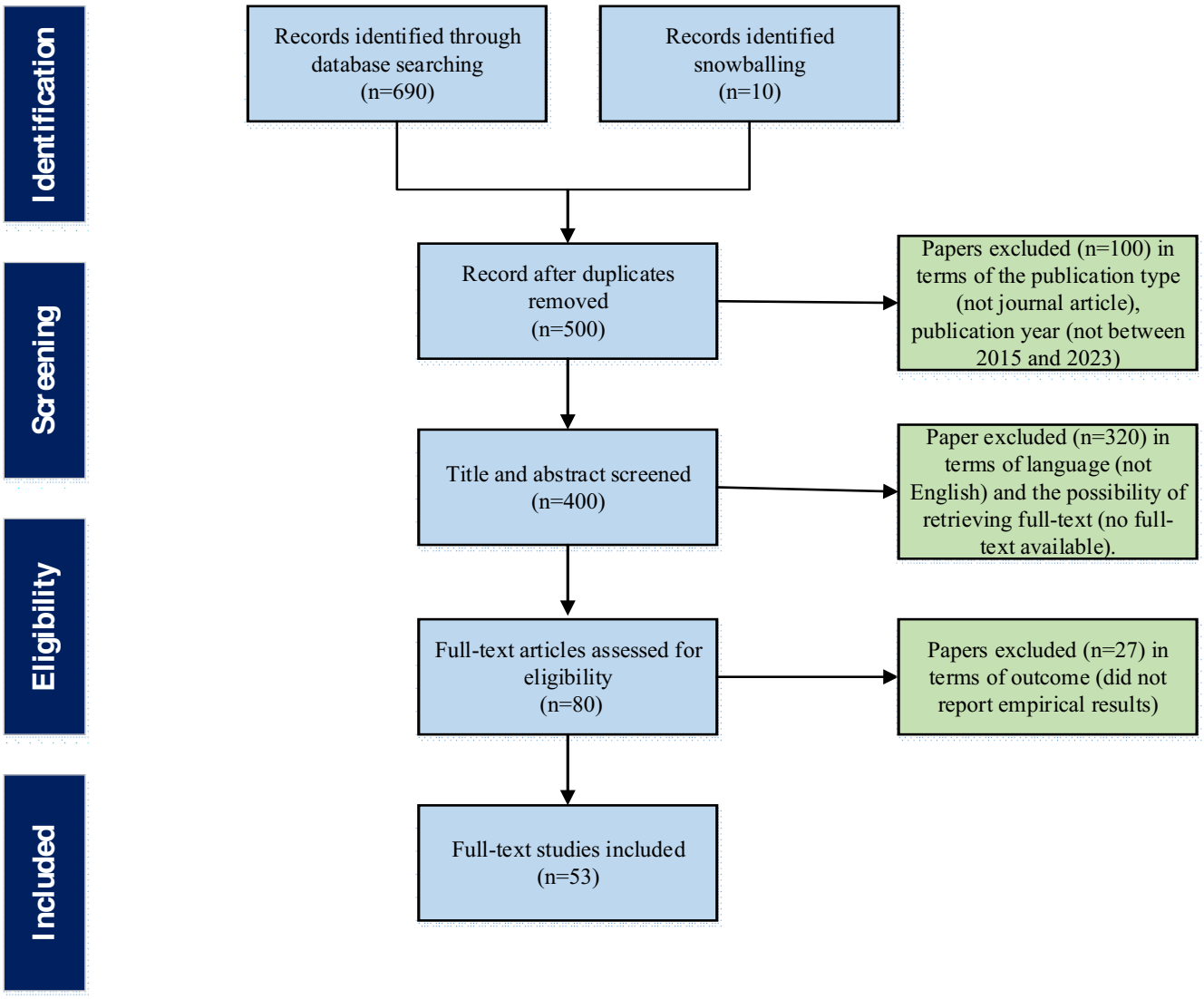


Figure 2.3 Systematic literature review of the impact of AI adoption on students' academic performance (Source: Author's work)

The prevalence of Artificial Intelligence (AI) is on the rise across multiple domains, including education. Specifically, AI adoption in online distance learning (ODL) settings offers several unique benefits and challenges. This review systematically examines the literature on this topic to explore the key factors driving AI adoption, the impact of these factors on academic performance, how these factors might be used to predict academic performance using a Support Vector Machine (SVM), and how moderating factors such as gender and regional differences can affect AI adoption's impact. The following four categories were employed in the systematic literature review to answer the research questions effectively:

I. Factors Driving AI Adoption in ODL
II. Impact of AI Adoption on Academic Performance in ODL
III. Predicting Academic Performance Using SVM
IV. Moderating Factors: Gender and Regional Differences

2.3.1 Factors Driving AI Adoption in ODL Settings

Research Question 1: What principal determinants propel AI adoption in Online Distance Learning (ODL) settings?

A comprehensive analysis was undertaken to discern the significant variables propelling the integration of Artificial Intelligence (AI) within Online Distance Learning (ODL) frameworks. Emerging from this study were several factors that played a pivotal role in driving this technological progression.

The leading technology acceptance theories, such as the Technology Acceptance Model (TAM), focus on ease of use and usefulness (Charness & Boot, 2016), the Information Systems Success (D&M Model), emphasizing system quality and user satisfaction (Sabeh et al., 2021), and the Unified Theory of Acceptance and Use of Technology (UTAUT), considering a broader framework including social influence (Yakubu & Dasuki, 2018), were reviewed. These theories offer a nuanced understanding of the factors affecting AI adoption in online distance learning (ODL) settings.

A primary catalyst of this trend is the potential for personalized and adaptive learning. The capacity of AI to customize educational paths to fit individual learners contributes to improved academic achievements and an uptick in student engagement (Bozkurt et al., 2021). Further facilitating this growth is adopting learning analytics, a tool that offers critical insights into student behaviours and learning styles. This amplifies the effectiveness of pedagogical feedback and refines teaching methodologies (Nguyen et al., 2020).

These conclusions are further substantiated by recent research. Studies by Almaiah et al. (2022) underscore the impact of enhanced academic outcomes, increased efficiency, cost-effectiveness, and tailored learning experiences on driving AI adoption. Furthermore, the research highlights the role of improved student engagement in promoting AI uptake in ODL contexts. Adding to this empirical evidence, Horowitz and Kahn (2021) affirm the importance of immediate feedback as a significant driving factor. Likewise, instructional quality, content relevance, motivation, and student relationships considerably influence student acceptance of ODL (Alam et al., 2022). From an organizational standpoint, compatibility, relative advantage, AI readiness, business process adaptability, and leadership have emerged as crucial for embracing AI (Kurup & Gupta, 2022).

At an individual level, perceived usefulness, performance expectancy, attitudes, trust, and effort expectancy shape AI technology's intention and actual usage (Kelly et al., 2022). Understanding these

variables, therefore, offers invaluable insights into the broader landscape of AI adoption in ODL contexts. Despite some overlaps, the key factors driving AI adoption within ODL and conventional educational settings vary considerably. Within ODL frameworks, aligning AI systems with the goals, values, and needs of institutions and students plays a vital role. Other significant factors include comparative benefits offered by AI over traditional education methods, the level of AI preparedness, and the capacity of institutions to adapt their processes to accommodate AI (Chen et al., 2020). The emotional dynamics of learning, encompassing learning-related anxiety and the readiness for online interaction and collaboration, also play a critical role. Additionally, the impact of AI systems on knowledge absorption and online interaction enhancement is considered integral to their adoption (Almaiah et al., 2022).

In contrast, AI adoption in traditional learning environments is influenced by different factors, such as performance anticipation, attitudes towards AI, the level of trust in the systems, the effort expectancy, and the perceived applicability of the technology (Kelly et al., 2022). In summary, the driving factors for AI adoption in ODL scenarios are primarily centred on aligning AI systems with institutional and learner needs and the capacity to adapt to technological advances. On the other hand, adopting AI in traditional learning settings leans more towards these systems' perceived usefulness and ease of use.

2.3.2 Impact of AI Adoption on Academic Performance in ODL

Research Question 2: How does AI adoption impact students' academic performance in ODL?

The influence of certain factors on students' academic performance in Open and Distance Learning (ODL) can be examined by addressing the research question above. Extensive literature suggests that the application of AI in ODL settings has a significant positive correlation with improvements in students' academic achievements. This effect is particularly pronounced when AI systems are utilized for personalizing learning and delivering timely, pertinent feedback (Zhu et al., 2018).

Moreover, AI-powered tools, such as intelligent tutoring systems, can provide custom-tailored instructions addressing individual student needs, leading to a marked enhancement in learning outcomes (Lu et al., 2021).

Several studies have reviewed the effect of these factors on students' academic performance and have highlighted their beneficial influence on academic achievement. The conclusions drawn from these studies emphasize the following mechanisms through which AI adoption positively affects academic performance:

- Enhancement of learning outcomes
- Boosting student engagement
- Provision of tailored learning experiences (Ali et al., 2023)
- Availability of immediate feedback (Bertl et al., 2022)

This evidence underscores the transformative potential of AI in the ODL landscape, revolutionizing the learning experience and driving educational success.

The factors driving the adoption of AI in ODL settings can positively impact academic performance. Here are some specific ways in which these factors can influence academic performance in ODL:

I. AI performance prediction models can accurately predict and monitor student academic performance in online higher education (Ouyang et al., 2023; Khan et al., 2021). This predictive capability helps identify at-risk students and establish student-centred learning pathways.
II. The integration of AI and learning analytics can improve student learning in online engineering courses (Ouyang et al., 2023). Providing students with in-time and continuous feedback enhances their learning quality.
III. AI-enabled prediction models can help anticipate academic achievement in online education, aiding instructors in preparing and delivering more effective teaching and learning (Jiao et al., 2022; Cruz-Jesus et al., 2020). This allows instructors to tailor their approaches to suit individual student needs.
IV. Machine learning algorithms can monitor students' academic progress and alert instructors about students at risk of unsatisfactory results in a course (Khan et al., 2021). Timely interventions can then be taken to improve student performance.
V. Machine learning algorithms can achieve high prediction accuracy and forecast student enrollment, college admission, dropout rates, and the risk of failure and withdrawal in online courses (Cruz-Jesus et al., 2020). This helps institutions support student success and improve decision-making.
VI. The integration of AI and learning analytics can support instructors in making informed decisions to facilitate student-centred learning and enhance the knowledge-construction processes of student groups (Ouyang et al., 2023).

The factors driving AI adoption in ODL can positively impact academic performance by accurately predicting and monitoring student performance, improving student learning, identifying students at risk of unsatisfactory results, and supporting instructors' informed decision-making. These results

emphasize the noteworthy capacity of artificial intelligence within open and distance learning to amplify academic achievements and generate a more efficient and individualized educational setting for students. However, it is important to note that implementing AI in ODL also presents potential disadvantages and challenges. Research has emphasized the potential drawbacks of implementing Artificial Intelligence (AI) in Open Distance Learning (ODL), with specific regard to students' academic performance (Almaiah et al., 2022). A fundamental observation is that learners' perceptions of AI can profoundly affect its success, indicating a necessity to lessen the anxiety associated with AI for better outcomes.

Notably, students and teachers share concerns about AI's role in education. They fear that overreliance on AI could unintentionally restrict students' chances for exploration and discovery (Seo, Tang, Roll, Fels, & Yoon, 2021). This concern is mirrored in their experiences, where many negative interactions with AI systems are rooted in misconceived expectations and misunderstandings about technology. Furthermore, adopting new technologies like AI often triggers anxiety, impeding their acceptance and use (Youmei Wang, Liu, & Tu, 2021). Another key concern is the risk of over-standardizing the learning process, which might diminish students' self-control in their learning paths. Although students acknowledge AI systems' potential aid, they also raise concerns that such standardized assistance could negatively impact their self-guided learning (Youmei Wang, Liu, & Tu, 2021).

Further inquiry is needed to comprehensively understand the impact of AI adoption on students' academic performance in ODL. While AI promises to enrich learning outcomes and promote personalized education, concerns remain regarding over-standardization, elevated anxiety, and potential adverse impacts on self-directed learning. A multifaceted challenge exists in deciphering the complex interplay of factors influencing AI adoption and its effects on academic performance. A comprehensive understanding of these factors and the development of appropriate frameworks will pave the way for effective and responsible use of AI in ODL, promoting educational success in the digital age. It is essential to develop a process framework that can predict the impact of AI adoption on students' academic performance in ODL, ensuring that it effectively enhances learning while addressing challenges related to over-standardization, anxiety, and self-directed learning. Such a process framework could provide an essential tool in anticipating and managing these outcomes, ensuring that AI's integration in ODL effectively enhances learning while addressing the associated challenges.

2.3.3 Predicting Academic Performance Using SVM

Research Question 3: How can these factors predict students' academic performance using the

Support Vector Machine (SVM) approach?

A plethora of research has been dedicated to applying Support Vector Machine (SVM) to predict student outcomes in online distance learning (ODL) environments. A notable example is a study by Mduma et al. (2019) that offers a holistic view of machine learning methodologies for predicting student dropout. It highlights using multiple machine learning models, including SVM, to predict student dropout and factor in demographics, academic records, and engagement levels. The study underscores the potential of machine learning as an identifier of at-risk students, providing an opportunity for targeted interventions to reduce dropout rates. Additionally, Tomasevic et al. (2020) suggest that SVM can effectively predict student performance when trained on relevant parameters like historical academic records, engagement metrics, and behavioural tendencies.

The potential of SVM in forecasting student performance has been revealed through further exploration, considering a variety of influencing factors. For instance, Ayouni et al. (2021) assessed the efficacy of machine learning algorithms for predicting student engagement in online learning environments. Their study found the SVM algorithm particularly effective in predicting engagement levels by analysing student interactions within the online learning platform. This suggests the potential of SVM as a tool to boost student engagement and improve overall learning outcomes.

Within online education, AI performance prediction models have demonstrated remarkable progress. They have been employed in online higher education to predict and monitor student performance by leveraging student learning data and machine learning algorithms (Ouyang et al., 2023). For instance, an AI-powered prediction model was developed to predict learning outcomes for students in online engineering education (Jiao et al., 2022). The integration of AI and learning analytics has sparked innovative pedagogical approaches. Such fusion offers educators a wealth of data to stimulate student-focused learning and strengthen knowledge-building within student cohorts. The insights obtained from this integration can significantly enhance the quality of online education (Ouyang et al., 2023).

The scope of AI extends beyond student learning to predict instructor performance. Xiao et al. (2021) proposed a model that comprehensively analyses numerical data associated with several teacher-relevant factors to assess instructor performance. This demonstrates the potential of AI in not only boosting student learning but also enhancing teaching methods. AI algorithms are vital in developing performance prediction models for online education. Machine learning, a subset of AI, is widely used to predict academic outcomes in digital learning environments (Jiao et al., 2022). By detecting

intricate patterns within data, these algorithms can forecast student performance accurately. Evolutionary computation, another facet of AI, has been employed to develop models that predict student performance in online learning contexts (Jiao et al., 2022). This approach, which mimics the processes of natural evolution, can solve complex optimization issues, thereby increasing prediction accuracy. AI algorithms, particularly machine learning, contribute substantially to constructing AI performance prediction models, utilizing student learning data to forecast and monitor academic progress (Ouyang et al., 2023; Jiao et al., 2022). These models also assess instructor performance, showcasing AI's extensive role in online education.

The efficacy of AI algorithms in performance prediction models for online education can vary based on the specific algorithm applied. Our findings reveal that machine learning algorithms are extensively used to predict academic performance in online educational contexts (Ouyang et al., 2023; Jiao et al., 2022; Holicza & Kiss, 2023). Notably, evolutionary computation has been employed to construct predictive models, as demonstrated by Jiao et al. (2022), who developed a student performance prediction model using this technique. A comparative study by Holicza & Kiss (2023) evaluated the efficacy of different machine learning algorithms in predicting online and offline student academic performance, with the Random Forest algorithm exhibiting the highest accuracy. Among various AI algorithms, the SVM approach has shown promising results. It is more accurate than other machine learning algorithms in predicting student performance (Ouyang et al., 2023). These results underscore the necessity of selecting the most suitable AI algorithm to enhance prediction accuracy in online education models. AI algorithms offer substantial benefits in predicting student performance in online education. They facilitate the early identification of at-risk students, enabling preventive measures to improve performance (Ouyang et al., 2023).

Additionally, AI algorithms provide personalized recommendations to enhance academic performance, and by dissecting individual student performance data, these algorithms can create tailored strategies to improve learning outcomes (Ouyang et al., 2023). Combining AI and learning analytics provides educators with crucial data for informed decision-making, fostering student-centred learning and refining knowledge-building processes (Ouyang et al., 2023). Lastly, AI's predictive analytics capacity can analyse student performance data to anticipate potential issues and forecast future outcomes, empowering educators to address academic challenges and proactively provide targeted support to students (Kelly et al., 2023). In essence, AI performance prediction models provide substantial advantages in online education. They accurately predict and monitor student performance, aiding in identifying at-risk students and crafting student-centric learning pathways. They also equip educators with the necessary data to improve performance and make

informed decisions. Furthermore, using different AI algorithms and their varying accuracy underscores the significance of choosing the most effective AI algorithm, like SVM, for precise performance prediction.

2.3.4 Moderating Factors: Gender and Regional Differences

Research Question 4: How do moderating factors such as gender and regional differences affect the impact of AI adoption on students' academic performance in ODL?

The research underscores the moderating role of factors like gender and regional disparities on the impact of AI adoption on students' academic performance in Online Distance Learning (ODL). For instance, notable discrepancies exist in the attitudes towards and usage patterns of AI-enhanced educational tools between male and female students (Gardner, Brooks & Baker, 2019). Additionally, regional variances, such as the availability of technological infrastructure and the prevailing cultural attitudes towards technology, can influence the effectiveness of AI in education (O'Dea & O'Dea, 2023). Kumar and Choudhury (2022) highlighted the issue of gender inequality within artificial intelligence. The development process of AI systems can inadvertently embed gender bias due to unconscious biases held by the algorithm developers. They may unknowingly transmit these socially ingrained biases to AI systems. This bias is exemplified in how current trends in machine learning reinforce age-old stereotypes about women, such as their perceived modesty, gentleness, and the need for protection. For instance, the majority of security robots are designed as male, while most service and sex robots are female.

Toplic (2021) emphasized that the growing ubiquity of AI carries profound implications. Barriers to accessing and using digital technologies, including AI, can hinder women and girls from leveraging opportunities in education, the economy, and society. Astonishingly, out of the world's 796 million illiterate individuals, over 66% are women. Furthermore, the majority of the world's 2.9 billion people without internet connectivity are women. Evidence shows that women are 25% less likely than men to possess digital proficiency for everyday tasks. This lack of equal access to digital technologies, including AI, obstructs women and girls' progress in economic, social, and educational domains. Therefore, understanding the factors driving gender differences in the adoption of AI-based applications in ODL settings is crucial for promoting gender inclusion and equality principles in the adoption of AI for sustainable education.

In conclusion, the systematic literature review indicates that AI is promising to enhance outcomes in Online Distance Learning (ODL). However, it is crucial to consider personal and regional differences

that can influence its effectiveness. Ongoing research should continue to explore these differences and construct tactics to optimize the advantageous aspects of Artificial Intelligence in Open and Distance Learning. Further investigation is required to fully understand the influence of moderating factors, such as gender and regional disparities, on the impact of AI adoption on students' academic performance in ODL. The present research aims to investigate the specific impact of these moderating factors on students' academic performance within the context of Online Distance Learning.

Overall, this systematic literature review provides valuable insights into the driving factors behind AI adoption in ODL and their impact on students' academic performance. The use of support vector machine (SVM) as a predictive model and the development of process frameworks show promise in predicting the effects of AI adoption on academic performance. The findings suggest that AI adoption has the potential to improve learning outcomes, enhance student engagement, and provide personalized learning experiences in ODL. However, further research is needed to explore the moderating factors that influence the impact of AI adoption, such as gender and geographical location differences. It is also important to address the limitations of the reviewed studies, including small sample sizes and limited generalizability, in future research. This systematic literature review is a foundation for further investigations into the factors influencing AI adoption and its ramifications on student academic performance in ODL settings. The study's outcomes can inform the development of effective strategies for promoting the successful integration of AI in education. It is essential to continue advancing research in this field to unlock the full potential of AI in enhancing outcomes in Online Distance Learning.

2.4 Review of Related Works

The potential impact of AI on education as a whole has been discussed in several studies (Chen et al., 2020; Shen et al., 2021; Chaudhry & Kazim, 2021; Khare, Stewart, & Khare, 2018; and Tanveer, Hassan, & Bhaumik, 2020). However, Ouyang, Zheng, and Jiao (2022) note that there is still a need for more empirical research to test the actual effects of AI applications in online higher education. Allam, Hassan, Mohideen, Ramlan, and Kamal (2020) highlight the limited research that focuses on the direct impact of AI adoption on students' academic performance, particularly in the context of ODL systems. While some studies have explored the use of AI in education, they have predominantly centred on traditional classroom settings and have not fully addressed the distinctive characteristics of ODL systems. The study also suggests that further research is needed to understand how the various factors influencing AI adoption affect academic performance, specifically in ODL systems. The application of AI in educational settings presents numerous opportunities, particularly for ODL institutions. Given that ODL heavily relies on human-machine interactions, AI offers these

institutions various avenues to address key aspects such as effective learning methods, teaching strategies, and the advantages and limitations of computer-based systems in education (Oyedeji, Salami, Folorunsho, & Abolade, 2020).

The study conducted by Allam, Hassan, Mohideen, Ramlan, and Kamal (2020) revealed a low level of self-directed learning and metacognitive online learning among undergraduate students, indicating the necessity for additional research to explore how artificial intelligence (AI) can support these areas. Tait (2014) emphasizes the need to reconfigure student support in the digital age, specifically in distance and e-learning, which involves understanding the impact of AI adoption. Chaudhary and Dey (2013) underscore the importance of diverse assessment techniques and methods in open and distance learning (ODL), including exploring AI's impact on assessment. Olivier (2016) investigates the influence of face-to-face contact sessions and virtual discussion forums on academic performance in ODL, further emphasizing the need for research to comprehend the impact of AI on academic achievement. Koneru (2017) discusses the significance of assessment in ODL for enhancing learning and monitoring academic program effectiveness, which necessitates understanding the impact of AI on assessment. Msweli (2012) recognizes ODL as an effective means of promoting educational equity, emphasizing the requirement for research on how AI can support this goal.

Furthermore, Khor (2014) analyzes student perception and adoption behaviour of ODL using the technology acceptance model, providing valuable insights into students' perspectives and adoption of AI in ODL. Rifin, Kadiran, and Bakar (2022) address the challenges faced by students and lecturers in transitioning from conventional lecture-based approaches to online distance learning, emphasizing the need for research on the impact of AI adoption on academic performance. Therefore, there is a clear need for further research to gain a comprehensive understanding of how AI adoption influences students' academic performance in ODL systems. These research endeavours contribute to advancing the integration of AI in education and its potential to enhance learning outcomes in ODL settings.

Therefore, the problem addressed by this research is the lack of a comprehensive framework that can predict the impact of AI adoption on academic performance in ODL systems. Given the increasing adoption of AI in ODL systems, there is a need to develop a process-based framework that can predict the impact of AI on academic performance. This research aims to develop a comprehensive process-based framework for predicting the impact of AI adoption on students' academic outcomes in Open and Distance Learning (ODL) using a Support Vector Machine (SVM), focusing on gender and regional differences. The main focus of this research is to design, validate, and implement the process framework, with the implementation phase involving the use of Support Vector Machine learning for

prediction. Additionally, the study evaluated the efficacy of the implemented system. A dataset was collected from ODL students to facilitate the prediction process to ensure accurate and reliable results. The study identified the factors impacting AI adoption that influence students' academic performance. By accounting for gender and regional differences in the proposed framework, the study promotes inclusive and equitable quality education through AI in ODL, which aligns with the United Nations' Sustainable Development Goal 4. The findings help design and implement effective AI-based interventions to enhance students' academic performance in ODL systems. Additionally, the research contributes to advancing knowledge of AI and its impact on education, particularly in the context of ODL systems. The predictive process framework and model will offer the following benefits:

- **Forecasting and Planning with Standardization:** Gathering data directly from students provides immediate insights into their experiences with AI tools and the effects on their academic performance through statistical analysis. However, developing a predictive model for assessing the impact of AI adoption on academic performance offers distinct advantages. This approach allows for a broader understanding of AI's potential effects before widespread implementation, enabling educators and policymakers to make informed decisions, tailor educational strategies, and anticipate long-term outcomes. Predictive modelling extends beyond immediate feedback, providing strategic, scalable, and efficient insights for enhancing the integration of AI in educational settings. By forecasting the potential impact of AI tools before their widespread implementation, educators and policymakers can engage in better planning and resource allocation. The process framework standardizes this approach, ensuring consistency across different contexts and enabling reliable comparisons and adjustments based on forecasted outcomes.
- **Insight into Variables and Best Practices Incorporation:** Predictive modelling identifies crucial factors affecting academic performance in AI, such as student engagement and the specifics of AI tool applications. The framework enhances this by ensuring that data collection and analysis follow best practices in data science, educational technology, and ethics, focusing on enhancing predictive model relevance and applicability.
- **Scalability and Iterative Improvement:** Direct data collection offers valuable insights but is not always scalable. A predictive model, underpinned by a process framework, can be broadly applied and continuously refined. This iterative improvement process ensures that models remain accurate and relevant as educational contexts and technologies evolve.
- Efficiency and Resource Optimization: Predictive models provide quicker assessments, allowing real-time educational strategy adjustments. The process framework underlines this efficiency by offering a clear roadmap for development, ensuring targeted and efficient resource allocation towards activities that significantly impact model quality and utility.

- **Customization and Stakeholder Engagement:** Understanding how students might respond to AI-based learning tools enables more personalized experiences. The process framework fosters stakeholder engagement, ensuring that predictive models reflect diverse needs and perspectives for more effective AI integration strategies.
- **Longitudinal Studies and Outcome Focus:** Predictive models simulate long-term AI adoption effects, which are crucial for sustainability and long-term benefits. The process framework ensures these efforts align with educational outcomes, focusing on initiatives significantly enhancing learning and teaching.
- **Cost-Effectiveness and Scalability:** Developing a predictive model is more cost-effective than continuous data collection and analysis. The framework emphasises this cost-effectiveness and facilitates model scalability and replicability across different educational settings, broadening AI's educational impact.
- **Transparency and Accountability:** A well-defined process framework increases the transparency of how predictive models are developed and used, helping to gain the educational community's trust and ensuring accountability in decision-making.

Incorporating a process framework for predicting the impact of AI adoption on academic performance thus not only enhances predictive modelling efforts but ensures that these initiatives are consistent, collaborative, transparent, and ultimately focused on improving educational outcomes. This cohesive approach leverages the strengths of both predictive modelling and structured frameworks to optimize the integration of AI into educational systems, ensuring that the adoption of AI tools is as effective and beneficial as possible.

This current research investigates the regional differences in the developed framework using West Africa (with Nigeria as a case study) and North America (with Canada as a case study). These regions' social, political, and economic structures vary significantly, making them ideal for comparative analysis. The project begins with a scoping review of existing literature, aiming to develop an extrapolative decision support system. A modified Machine Learning (ML) algorithm improves model accuracy. Specifically, Support Vector Machine (SVM) is utilised to analyze the complex relationship between AI adoption and academic performance. The SVM model is explored and potentially modified to enhance its accuracy depending on the characteristics of the datasets. Previous studies have demonstrated that different approaches to the same problem can yield varying outcomes (Nourani, Gökçekuş, and Umar, 2020). This research aims to minimize error variance and produce more reliable results than traditional models (e.g., Structural Equation Model or Statistical Method) used in the eLearning domain by employing a modified ML algorithm.

The study's findings help identify the factors influencing AI adoption and gender differences in AI application adoption in Open and Distance Learning (ODL) settings. This understanding is valuable for developers, higher education providers, policymakers, and the government in promoting gender inclusion and meeting students' needs through AI-based application platforms. To ensure the ethical integrity of the research, it underwent scrutiny by an Ethics committee and received approval from the National Open University's Faculty Research Ethics Committee (FREC). Adhering to the fundamental ethical principles of human subject protection, including respect for persons, beneficence, and justice, will be paramount throughout the research process.

Over the last few decades, the intersection of artificial intelligence (AI) and education has garnered considerable attention from researchers across disciplines. As Hwang et al. (2020) indicated, advances in computing and data processing techniques have expedited AI development, primarily intending to mimic intelligent human behaviour such as inference, analysis, and decision-making. The anticipation is to witness a surge of research focusing on how AI can be seamlessly integrated into classrooms and how AI expertise can be imparted to students across different educational levels. Chaudhry and Kazim (2021) offered a recent overview of AI in Education (AIEd) research, underlining its potential to reduce teachers' burden, personalise learning experiences, revolutionise assessments, and contribute to intelligent tutoring systems' progression. The study suggests that the central thesis of AIEd is to bolster education rather than merely promote AI. Groundbreaking AI from international researchers and businesses is valuable only if it aids students in their learning journeys. Thus, learning outcomes are the ultimate yardstick for evaluating AI's impact on education.

In this regard, Hwang et al. (2020) have underscored several research directions in AIEd, including scrutinizing AI-based learning systems' efficacy from diverse viewpoints. Four primary domains surfaced where AI applications in education were found: profiling and prediction, assessment and evaluation, adaptive systems and personalization, and intelligent tutoring systems. These applications predominantly reside within academic support services and institutional and administrative services. The vast majority of AI in education literature is situated within computer science and STEM fields, and empirical investigations mainly employ quantitative methodologies such as the structural equation modelling approach. This present work investigates AIEd's role in supporting education by studying its applications in Online Distance Learning (ODL) settings. This includes exploring the factors that stimulate its adoption, the impacts of these factors on students' academic performance, and the role of gender and regional differences in its adoption. The research employed quantitative methodologies, specifically Machine Learning Modelling, to examine these factors in ODL settings

in West Africa and North America.

Popenici and Kerr (2017) studied AI systems' influence on learning and teaching, unearthing potential discord between learners and educators. Their work accentuates the necessity to comprehend AI systems' impact on learner-educator interactions in the online learning milieu. Roll and Wylie (2016) advocated for the increased involvement of AI systems in learner-educator communication and educational applications beyond the school context, suggesting that AI systems could significantly enhance online learner-educator interactions. Demir and Yurdugül's teacher readiness model encompasses eight critical components: acceptance, technology access, motivation, self-efficacy, perceived ease of use, perceived usefulness, perceived enjoyment, and social influence. These factors are considered fundamental for adopting online distance learning (Demir and Yurdugül, 2015).

The technology acceptance model (TAM) has evolved into a theoretical framework for using and accepting online technologies. Muhaimin et al. (2019) suggest that these models rely on a range of concepts, including attitude towards technology, perceived ease of use of the technology, and perceived usefulness of the technology. Muhaimin et al. (2019) conducted a study during the COVID -19 pandemic in Malaysia to evaluate the factors influencing the intent to use online distance learning technology. They discovered a significant impact of perceived ease of use, perceived usefulness, and attitude towards technology on the intent to use online distance learning technology.

The AI in Education (AIEd) community is increasingly scrutinizing the impact of AI systems on online education. Uunona and Goosen (2023) noted that AI and machine learning have substantial potential to transform educational institutions. An abundance of scholarly work exists concerning implementing AI in education, especially in the context of Online Distance Learning (ODL) (Picciano, 2017; Haenlein & Kaplan, 2019). The promise of AI delivering personalized and adaptive learning is a powerful catalyst for its uptake, accompanied by advantages like improved efficiency and heightened student engagement, as mentioned in various studies (Tiwari, 2023; Hashim et al., 2022).

The principal interest of this research is to investigate the effect of AI on learners' academic achievement. Numerous research studies support the hypothesis that AI's personalized learning and prompt feedback significantly bolster students' academic performance in ODL contexts (Zhu et al., 2018; Akyuz, 2020). Further, implementing intelligent tutoring systems has been associated with enhanced learning outcomes (Akyuz, 2020; Ali et al., 2023). Existing research offers encouraging results in forecasting student performance using a Support Vector Machine (SVM). This exploration

uses machine learning techniques, specifically SVM, to predict student outcomes in online learning by considering past performance, engagement metrics, and behavioural patterns. SVM has proven effective in forecasting student results in ODL settings (Alqahtani, 2021). For instance, one study illustrated SVM's capacity to predict student engagement levels, which are crucial predictors of academic achievement. Another study examined the prediction of student academic achievement during online learning utilizing regression in SVM. Factors such as attendance, participation, and quiz scores were used to predict academic achievement, with results demonstrating SVM's high efficiency in this task (Samsudin et al., 2022).

Academic performance prediction is a crucial aspect of online education as it helps identify students at risk of failure, enables personalized learning pathways, and optimizes instructional design (Asif et al., 2017; Chen et al., 2020; Roll and Wylie, 2016). Various AI algorithms have been employed in previous studies to predict students' examination performance using classification and regression techniques (Tomašević et al., 2020). For instance, researchers have used multiple machine learning techniques, such as Naïve Bayes and k-nearest neighbours, to categorize students as "pass" or "fail" ((Jiao et al., 2022). Other studies have explored learning algorithms to classify student results into different categories, including "pass" or "fail," high, middle, and low levels, and multiple classes based on achieved grades (Sandra et al., 2021). Some research has focused on predicting student failure or developing early warning systems using genetic programming and data mining algorithms (Nagy & Molontay, 2023; Jiao et al., 2022).

Wang, Liu, and Tu (2021) employed a structural equation modelling (SEM) approach to investigate teachers' continued intention to teach with AI. They examined factors such as anxiety, self-efficacy, attitude towards artificial intelligence (AI), perceived ease of use (PEU), and perceived usefulness (PU). The study aimed to understand the interactions among these factors and their influence on teachers' intention to use AI in their teaching. The research involved 311 higher education professors, and the SEM analysis revealed that PU, PEU, self-efficacy, and attitude towards AI explained a significant portion of the variation in teachers' behavioural intention. Attitude towards AI had the most substantial impact, followed by self-efficacy. The study found a positive relationship between teachers' self-efficacy and the adoption of AI-based applications, which, in turn, influenced PEU, attitude towards AI, and PU.

Interestingly, a negative correlation was observed between teachers' self-efficacy and their attitudes towards using AI-based applications. This suggests that enhancing self-efficacy could reduce reluctance to adopt such applications in teaching. The study utilized the SEM approach, and its results

were compared with machine learning modelling methods. Besides classification and regression, AI-enabled prediction models have been developed to forecast academic performance based on specific input variables characterizing student learning. These models can be categorized into similarity-based, model-based, and probabilistic approaches (Tomašević et al., 2020). However, there are gaps in the current development of prediction models concerning data identification and analytics. Many studies consider various student information data, such as demographics, without explicitly focusing on variables that reflect the specific learning process (Kurniawan et al., 2022). To address this issue, researchers should deliberately select student data aligned with learning theories and the principles of student-centred learning. Promisingly, emerging studies are exploring process-oriented online learning behaviour data to accurately predict academic performance, moving beyond traditional student information or performance data (Bernacki et al., 2020). This research project designs a collaborative learning mode in online courses that aligns with this trend. It deliberately selects student data from the collaborative process to make accurate academic performance predictions.

A review by Manhica, Santos, and Cravino (2022) provides an overview of AI applications in learning management systems (LMS) within higher education. The review found that Moodle is the most popular LMS for implementing AI solutions, and AI modelling has been extensively used to assess student performance. This review also emphasises exploring the moderating factors influencing AI adoption, such as gender and regional differences. Although a dearth of literature directly addresses gender differences in AI adoption, extant studies imply that gender-based biases may unintentionally find their way into AI systems, possibly influencing user interactions and academic results (Daraz et al., 2022). Concerning regional differences, variations in AI adoption rates are apparent, probably due to disparities in technology infrastructure and cultural perceptions of technology (Pillai & Sivathanu, 2020). The study discusses factors shaping the adoption of AI technology across different regions, like the availability of infrastructure, cultural attitudes towards technology, and economic impacts. These factors can differ geographically and affect the rates of AI technology adoption.

Overall, these related works shed light on the driving factors behind AI adoption in online distance learning environments and their impacts on students' academic performance. They suggest AI systems' potential to enrich learner-educator interaction in online learning. The SVM approach can predict students' academic performance, considering factors like acceptance, technology access, motivation, self-efficacy, perceived ease of use, usefulness, enjoyment, and social influence. Furthermore, gender and regional differences as moderating factors can be considered to comprehend better AI adoption's impact on students' academic performance.

2.5 Summary/Meta-Analysis of Reviewed Related Works

2.5.1 Summary

The intersection of artificial intelligence (AI) and education has been a significant area of study over recent years, with researchers focusing on integrating AI into classrooms and imparting AI knowledge to students at all educational levels. Computational advancements have propelled AI's development, aiming to emulate human intelligence, including inferential and analytical capabilities (Hwang et al., 2020). AI's role in education (AIEd) has been explored comprehensively, highlighting its potential to alleviate teaching burdens, personalise educational experiences, transform assessments, and further the growth of intelligent tutoring systems (Chaudhry and Kazim, 2021). The crux of AIEd research is on enhancing education, with the impact on learning outcomes serving as the primary metric to assess the effect of AI in education.

Several research directions in AIEd have been proposed, which include examining the efficiency of AI-based learning systems. The applications of AI in education are typically observed within four primary areas: profiling and prediction, assessment and evaluation, adaptive systems and personalization, and intelligent tutoring systems. These applications are most commonly applied within academic support and administrative services. Most AIEd literature is found within Computer Science and STEM fields, primarily employing quantitative research methods, such as the Structural Equation Modeling Approach (Hwang et al., 2020).

The impact of AI on teaching and learning has been studied, revealing potential conflicts between teachers and learners (Popenici and Kerr, 2017). Roll and Wylie (2016) advocated for increased AI involvement in learner-educator communication and educational contexts beyond the classroom, suggesting that AI could substantially enhance online learner-educator interactions.

The adoption of online distance learning is influenced by several factors, such as acceptance, access to technology, motivation, self-efficacy, perceived ease of use, perceived usefulness, enjoyment, and social influence, as described in Demir and Yurdugül's teacher readiness model (Demir and Yurdugül, 2015). Similarly, the technology acceptance model (TAM) has emerged as a theoretical framework for the acceptance and usage of online technologies, which incorporates concepts such as perceived ease of use, perceived usefulness, and attitudes towards technology (Muhaimin et al., 2019).

The impact of AI systems on online education has been a growing focus for the AIEd community. AI and machine learning have demonstrated substantial potential to revolutionize educational institutions, particularly in Online Distance Learning (ODL) (Uunona and Goosen, 2023).

Predicting student performance using a Support Vector Machine (SVM) has shown promising results

in existing research. This technique uses machine learning to predict student outcomes in online learning by considering past performance, engagement metrics, and behavioural patterns. One study demonstrated the effectiveness of SVM in predicting student engagement levels, which are crucial predictors of academic achievement (Alqahtani, 2021). Gender and regional differences are significant moderating factors in AI adoption. While there is limited research on gender differences in AI adoption, some studies suggest gender-based biases may unintentionally be integrated into AI systems, potentially influencing user interactions and academic outcomes. Regional differences in AI adoption rates are likely due to disparities in technology infrastructure and cultural attitudes towards technology (Daraz et al., 2022; Pillai & Sivathanu, 2020).

Upon analysis of the related works in the field of artificial intelligence (AI) in education, a few key themes and findings become apparent:

I. **AI Development and Integration**: As highlighted by Hwang et al. (2020), the advancements in computing and data processing techniques have expedited the development of AI, with the primary focus on mimicking intelligent human behaviour. AI integration into classrooms and curricula is anticipated to surge thanks to its potential for inference, analysis, and decision-making.

II. **AI's Role in Education**: Studies like the one conducted by Chaudhry and Kazim (2021) illustrate AIEd's potential in several areas, including reducing teachers' workload, personalizing learning experiences, revolutionizing assessments, and enhancing intelligent tutoring systems. Moreover, the effectiveness of AIEd is ultimately measured by its impact on learning outcomes, which bolsters education rather than merely promoting AI.

III. **AI Applications in Education**: According to Hwang et al. (2020), AIEd's primary domains include profiling and prediction, assessment and evaluation, adaptive systems and personalization, and intelligent tutoring systems. These applications primarily support academic services and administrative services. The methodologies in these domains predominantly employ quantitative research methods such as the Structural Equation Modeling approach.

IV. **Learner-Educator Interactions**: The works of Popenici and Kerr (2017) and Roll and Wylie (2016) emphasize the importance of understanding the impact of AI systems on the dynamics between learners and educators. They advocate for more AI involvement in enhancing online learner-educator interactions.

V. **Acceptance of Online Distance Learning**: Demir and Yurdugül's (2015) model and the technology acceptance model (TAM) proposed by Muhaimin et al. (2019) highlight critical components influencing the acceptance of online distance learning technology, including

acceptance, access to technology, motivation, self-efficacy, perceived ease of use, and usefulness, enjoyment, and social influence.

VI. **AI's Impact on Online Education**: Uunona and Goosen (2023) and others discuss the transformative potential of AI and machine learning in educational institutions, specifically in Online Distance Learning (ODL).

VII. **Predicting Student Performance**: Using a Support Vector Machine (SVM) to predict student performance has shown promising results in the studies analyzed. The focus is on past performance, engagement metrics, and behavioural patterns (Alqahtani, 2021; Samsudin et al., 2022).

VIII. **Moderating Factors in AI Adoption**: The review identifies gender and regional differences as important moderating factors in AI adoption. Gender-based biases in AI systems, disparities in technology infrastructure, and cultural attitudes towards technology across regions are noteworthy (Daraz et al., 2022; Pillai & Sivathanu, 2020).

2.5.2 Meta-analysis

The included fifty-three articles are given in Table 2.1. The articles that addressed factors driving AI adoption, the impact of AI adoption on academic performance in ODL, the use of Support Vector Machine (SVM) for predicting academic performance, and the moderating factors of gender and regional differences were considered. The selection was based on the extent to which the articles significantly contributed to understanding these aspects and provided valuable insights and findings related to AI adoption and its impact on academic performance in ODL.

In this review, Machine Learning Methods were the most commonly used, accounting for 28.3% of the studies. This shows a significant interest in using machine learning techniques in the field and suggests that future research will continue to leverage these techniques to gain insights. Classical Statistical Methods were used in 22.6% of the studies. Although these methods might not be as cutting-edge as machine learning, they still play a crucial role in many research studies. Hybrid Methods were used in a very small proportion of the studies, specifically 3.8%. Non-empirical methods were used in 45.3% of the studies, making this the largest category. These methods include theoretical analyses, literature reviews, and other non-data-driven approaches. The distribution of the methodology used in the included studies is shown in Table 2.2. The chart that represents the methodology of the selected studies is shown in Figure 2.4. The chart illustrates the distribution of different research methods across the selected studies. The temporal distribution of articles included in this review (See Figure 2.5) reveals significant insights into the progression of research within the field.

A total of 53 articles spanning from 2015 to 2023 were analyzed. The number of articles published each year exhibited a general trend of increase over this period. A solitary article was published in 2015 and 2016. The year 2017 saw a modest increase with three articles. The figure dropped slightly to one in 2018, then increased to four in 2019. The year 2020 marked a substantial increase in publications, with six published articles. This upward trend continued into 2021 with a notable surge to 12 publications. The year 2022 saw the peak of this trend, with the highest number of articles - 16 - being published in a single year.

Table 2.1 The studies included in the final selection

S/N	Year of Publication	Articles	Methodology of the study
1	2015	Demir and Yurdugül, 2015	Non-Empirical Method
2	2016	Roll and Wylie, 2016	Non-Empirical Method
3		Popenici and Kerr, 2017	Non-Empirical Method
4	2017	Asif et al., 2017	Machine Learning Method
5		Picciano, 2017	Non-Empirical Method
6	2018	Zhu et al., 2018	Non-Empirical Method
7		Gardner et al., 2019	Statistical Method
8	2019	Mduma et al., 2019	Non-Empirical Method
9		Muhaimin et al., 2019	Statistical Method
10		Haenlein & Kaplan, 2019	Non-Empirical Method
11		Nguyen et al., 2020	Non-Empirical Method
12		Tomasevic et al., 2020	Machine Learning Method
13	2020	Bernacki et al., 2020	Machine Learning Method
14		Hwang et al., 2020	Non-Empirical Method
15		Chen et al., 2020	Non-Empirical Method
16		Akyuz, 2020	Statistical Method
17		Horowitz and Kahn, 2021	Statistical Method
18		Khan et al., 2021	Machine Learning Method
19		Seo, Tang, Roll, Fels, & Yoon, 2021	Statistical Method
20		Youmei Wang, Liu, & Tu, 2021	Statistical Method
21	2021	Alqahtani, 2021	Machine Learning Method
22		Wang, Liu, and Tu, 2021	Machine Learning Method
23		Chaudhry and Kazim, 2021	Non-Empirical Method
24		Ayouni et al., 2021	Machine Learning Method
25		Toplic, 2021	Non-Empirical Method

Table 2.1 The studies included in the final selection (Contd.)

S/N	Year of Publication	Articles	Methodology of the study
26		Sandra et al., 2021	Machine Learning Method
27	**2021**	Huang et al., 2021	Non-Empirical Method
28		Kuleto et al., 2021	Non-Empirical Method
29		Almaiah et al., 2022	Statistical Method
30		Kurup & Gupta, 2022	Statistical Method
31		Alam et al., 2022	Statistical Method
32		Bertl et al., 2022	Non-Empirical Method
33		Jiao et al., 2022	Machine Learning Method
34		Cruz-Jesus et al., 2020	Hybrid
35		Xiao et al., 2021	Machine Learning Method
36	**2022**	Hashim et al., 2022	Non-Empirical Method
37		Samsudin et al., 2022	Machine Learning Method
38		Manhica, Santos, and Cravino, 2022	Non-Empirical Method
39		Daraz et al., 2022	Non-Empirical Method
40		Pillai & Sivathanu, 2022	Hybrid
41		Kurniawan et al., 2022	Statistical Method
42		Liu & Huang, 2022	Statistical Method
43		Ogunsola-Bandele & Kennepohl, 2022	Statistical Method
44		Gao, 2022	Machine Learning Method
45	**2023**	Ouyang et al., 2023	Machine Learning Method
46		Holicza & Kiss, 2023	Machine Learning Method
47		Ali et al., 2023	Non-Empirical Method
48		Tiwari, 2023	Non-Empirical Method
49		Nagy and Molontay, 2023	Machine Learning Method
50		Uunona and Goosen, 2023	Non-Empirical Method
51		O'Dea & O'Dea, 2023	Non-Empirical Method
52		Tanjga, 2023	Non-Empirical Method
53		de la Torre-López, Ramírez, & Romero, 2023	Non-Empirical Method

Table 2.2 Methodology of the selected studies

S/N	Method of Research	Number of Articles	Percentage
1	Machine Learning Method	15	28.30%
2	Classical Statistical Method	12	22.64%
3	Hybrid Method	2	3.77%
4	Non-Empirical Method	24	45.28%

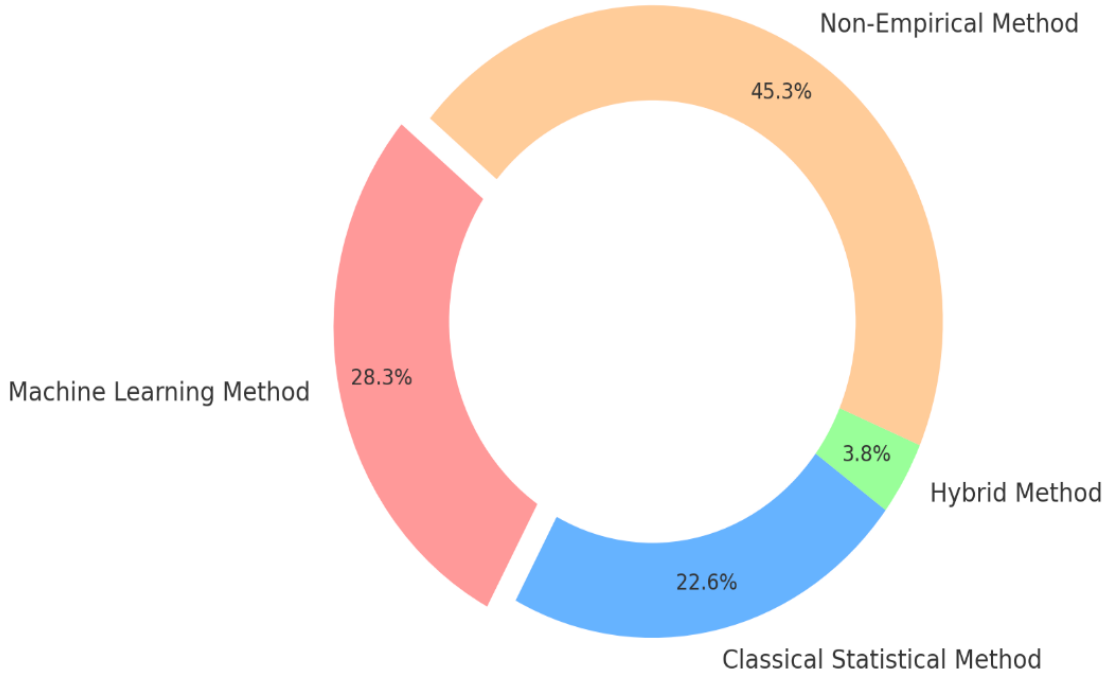


Figure 2.4 Percentage Distribution of the Method Used

However, in 2023, there was a slight decrease in the number of publications, with nine published articles. This dip could be attributed to the fact that the year was not yet over at the time of this review, or it could signal a new trend in the distribution of articles. This temporal distribution suggests a growing interest in the field, as reflected by the increasing number of articles published yearly. It also implies that the topics addressed by these articles are gaining traction in the research community, leading to a proliferation of studies and published works. This increasing trend in publication volume over the years points to the growing relevance and importance of this field and the need for continued research to keep pace with its rapid development. As such, the findings of this review are timely and pertinent to the current state of the field.

In this analysis, the journals mentioned in the dataset were examine based on their SCImago Journal Rank (SJR) and impact factor values. These metrics are widely used to assess the significance and influence of academic journals within their respective fields. Table 2.3 provides a list of articles along with the journals they were published in, the SCImago Journal Rank (SJR) for those journals and the impact factor of the journals. The high Impact Factors and SJRs of some of these journals indicate that the articles have been published in reputable journals and have a high potential for being cited in other works, which adds credibility to the articles. Here are some key insights from the provided data: **Journal Preference:** The journal 'Computers & Education' seems to be a popular choice for publication, with multiple entries listed. This suggests the journal's relevance and importance in the field of study.

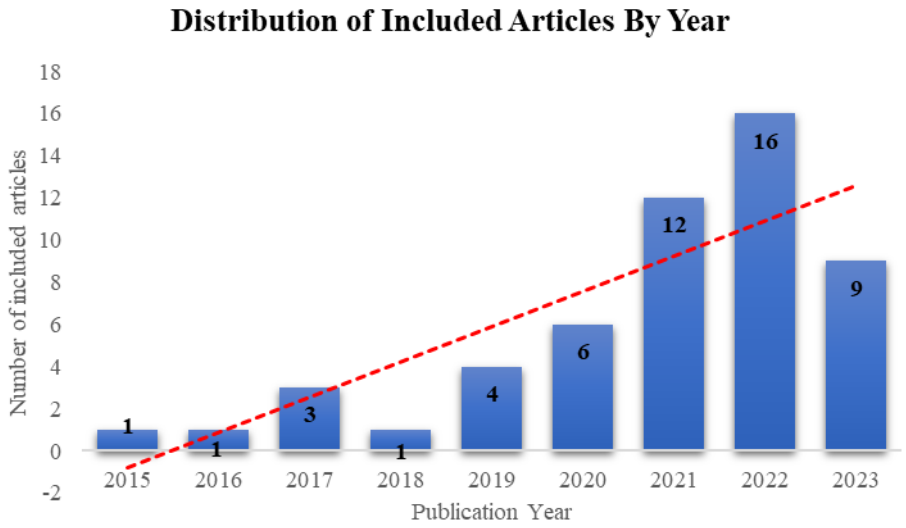


Figure 2.5 Distribution by year of the Articles included in the study

Journal Metrics: Generally, a higher SJR and Impact Factor are desirable as they suggest a more influential journal in the field. Notably, 'The Journal of Innovation & Knowledge' holds the highest SJR (2.649) and the highest Impact Factor (20.310) among the listed journals. This indicates the high recognition and influence of this journal. Figures 2.6-2.7 visually represent the top 10 journals by Impact Factor and SJR. 'Computers & Education' and 'Journal of Innovation & Knowledge' stand out in their respective categories, which supports the written findings.

Variation in Metrics: There is a wide variation in both SJR and Impact Factor across the different journals. This suggests a broad range of influence and reach for the listed journals.

Journals with Missing SJR and Impact Factor Information:

Some journals, including the International Journal of Progressive Education, International Learning Analytics & Knowledge Conference (LAK19), and Proceedings of the 53rd Hawaii International Conference on System Sciences, do not have an SJR or Impact Factor listed. This absence of data, marked as 'N/A', could stem from various reasons. For instance, these metrics might not be available for certain journals or conference proceedings. Alternatively, these could be relatively new or specialized journals for which such metrics have not yet been established. The lack of SJR and Impact Factor values makes it challenging to assess these journals' relative influence and reach within the academic community, at least through these particular metrics. However, it is essential to note that the evaluation of journals should not solely rest on these two metrics. The interpretation of SJR and Impact Factor values should be contextualized within the specific field or discipline of the journals.

In addition to these metrics, other factors like the journal's scope, the quality of the research it publishes, and its relevance to the research topic should also be considered when assessing the significance of a journal for a thesis or research study. This multi-faceted approach towards evaluation ensures a comprehensive understanding of the journal's standing and contribution to the field of study.

Table 2.3 Studies ranking and published journals

S/N	Articles	Journal Name	Scopus-SCImago Journal Rank (SJR)	Impact factor
1	Demir and Yurdugül, 2015	International Journal of Progressive Education	N/A	1.100
2	Roll and Wylie, 2016	Journal of Learning Analytics	1.369	4.760
3	Popenici and Kerr, 2017	Research and Practice in Technology Enhanced Learning.	0.654	3.440
4	Asif et al., 2017	Computers & Education	3.676	11.182
5	Picciano, 2017	Online Learning	1.417	5.030
6	Zhu et al., 2018	International Journal of Emerging Technologies in Learning	0.536	3.270
7	Gardner et al., 2019	International Learning Analytics & Knowledge Conference (LAK19)	N/A	N/A
8	Mduma et al., 2019	Data Science Journal	1.026	2.780
9	Muhaimin et al., 2019	Journal of Baltic Science Education	0.478	1.480
10	Haenlein & Kaplan, 2019	California Management Review	3.793	11.678
11	Nguyen et al., 2020	Proceedings of the 53rd Hawaii International Conference on System Sciences	N/A	N/A
12	Tomasevic et al., 2020	Computers & Education	3.682	15.58
13	Bernacki et al., 2020	Computers & Education	3.682	15.58
14	Hwang et al., 2020	Computers & Education	3.682	15.58
15	Chen et al., 2020	Computers & Education	3.682	15.58
16	Akyuz, 2020	Creative Education	N/A	0.500
17	Horowitz and Kahn, 2021	PLoS ONE	0.885	3.750
18	Khan et al., 2021	Smart Learning Environments	0.967	6.310
19	Seo, Tang, Roll, Fels, & Yoon, 2021	International Journal of Educational Technology in Higher Education,	2.051	10.42
20	Youmei Wang, Liu, & Tu, 2021	Educational Technology & Society	1.049	5.080
21	Alqahtani, 2021	Journal of Educational Computing Research	1.673	7.350
22	Wang, Liu, and Tu, 2021	Educational Technology & Society	1.049	5.080
23	Chaudhry and Kazim, 2021	AI And Ethics	N/A	N/A
24	Ayouni et al., 2021	PLoS ONE	0.885	3.750
25	Toplic, 2021	NetHope	N/A	N/A
26	Sandra et al., 2021	TEM Journal	0.231	1.210
27	Huang et al., 2021	Academic Journal of Interdisciplinary Studies	0.183	0.810
28	Kuleto et al., 2021	Sustainability	0.664	4.390
29	Almaiah et al., 2022	Electronics	0.148	0.530
30	Kurup & Gupta, 2022	A Journal of Management Research	0.567	3.460

Table 2.3 Studies ranking and published journals (Contd.)

S/N	Articles	Journal Name	Scopus-SCImago Journal Rank (SJR)	Impact factor
31	Alam et al., 2022	Education and Information Technologies.	1.249	7.65
32	Bertl et al., 2022	Frontiers in Psychiatry	1.222	4.52
33	Jiao et al., 2022	Artificial Intelligence Review	2.490	15.010
34	Cruz-Jesus et al., 2020	Heliyon	0.609	4.45
35	Xiao et al., 2021	Journal of Interconnection Networks	0.207	0.55
36	Hashim et al., 2022	International Journal of Academic Research in Progressive Education and Development	N/A	N/A
37	Samsudin et al., 2022	International Journal of Information and Education Technology	0.243	1.69
38	Manhica, Santos, and Cravino, 2022	2022 17th Iberian Conference on Information Systems and Technologies (CISTI).	0.146	0.493
39	Daraz et al., 2022	Computer and Information Science	0.924	6.053
40	Pillai & Sivathanu, 2022	Benchmarking	1.185	7.970
41	Kurniawan et al., 2022	Jurnal Pendidikan: Teori, Penelitian, Dan Pengembangan	N/A	N/A
42	Liu & Huang, 2022	Mathematical Problems in Engineering.	0.355	2.100
43	Ogunsola-Bandele & Kennepohl, 2022	In Tenth Pan-Commonwealth Forum on Open Learning.	N/A	N/A
44	Gao, 2022	Mathematical Problems in Engineering	0.355	2.100
45	Ouyang et al., 2023	International Journal of Educational Technology in Higher Education	2.051	10.420
46	Holicza & Kiss, 2023	Behav Sci (Basel)	0.597	2.980
47	Ali et al., 2023	Journal of Innovation & Knowledge	2.649	20.310
48	Tiwari, 2023	Indian Scientific Journal of Research in Engineering and Management	N/A	N/A
49	Nagy and Molontay, 2023	International Journal of Artificial Intelligence in Education.	1.110	4.980
50	Uunona and Goosen, 2023	In Advances in medical education, research, and ethics (AMERE)	N/A	N/A
51	O'Dea & O'Dea, 2023	Journal of University Teaching and Learning Practice	0.488	2.03
52	Tanjga, 2023	Qeios.	N/A	N/A
53	de la Torre-López, Ramírez, & Romero, 2023	Computing	0.824	4.331

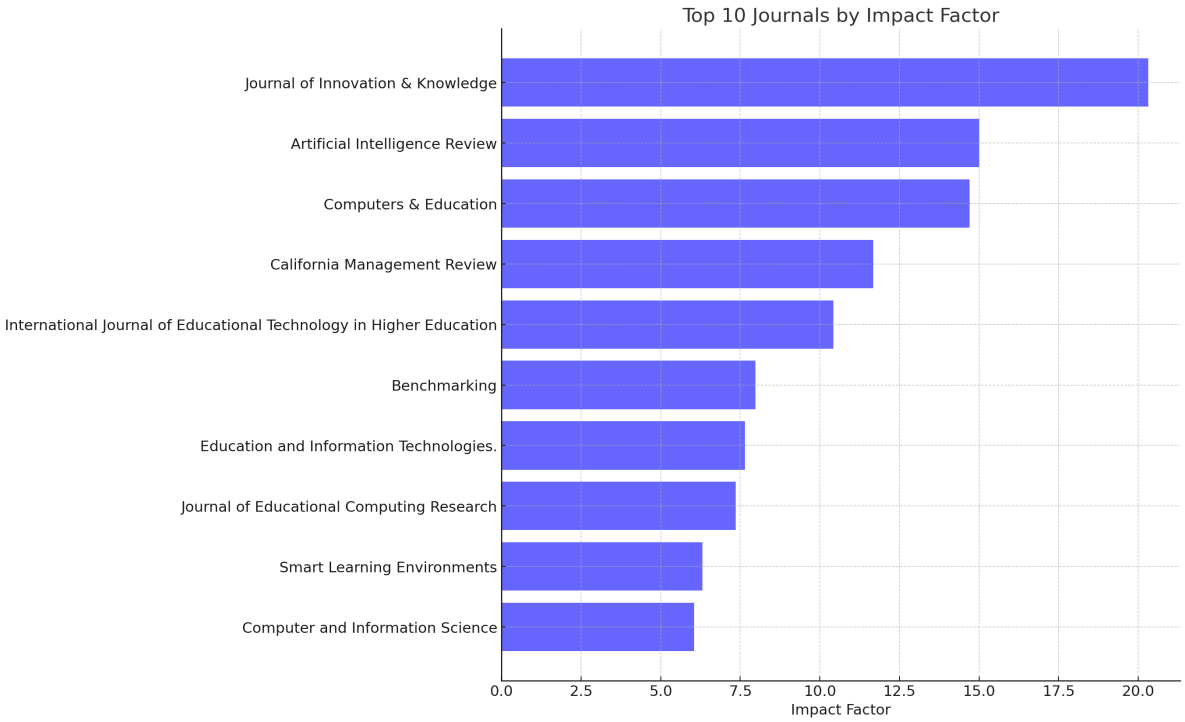


Figure 2.6 The Top 10 Journals by Impact Factor

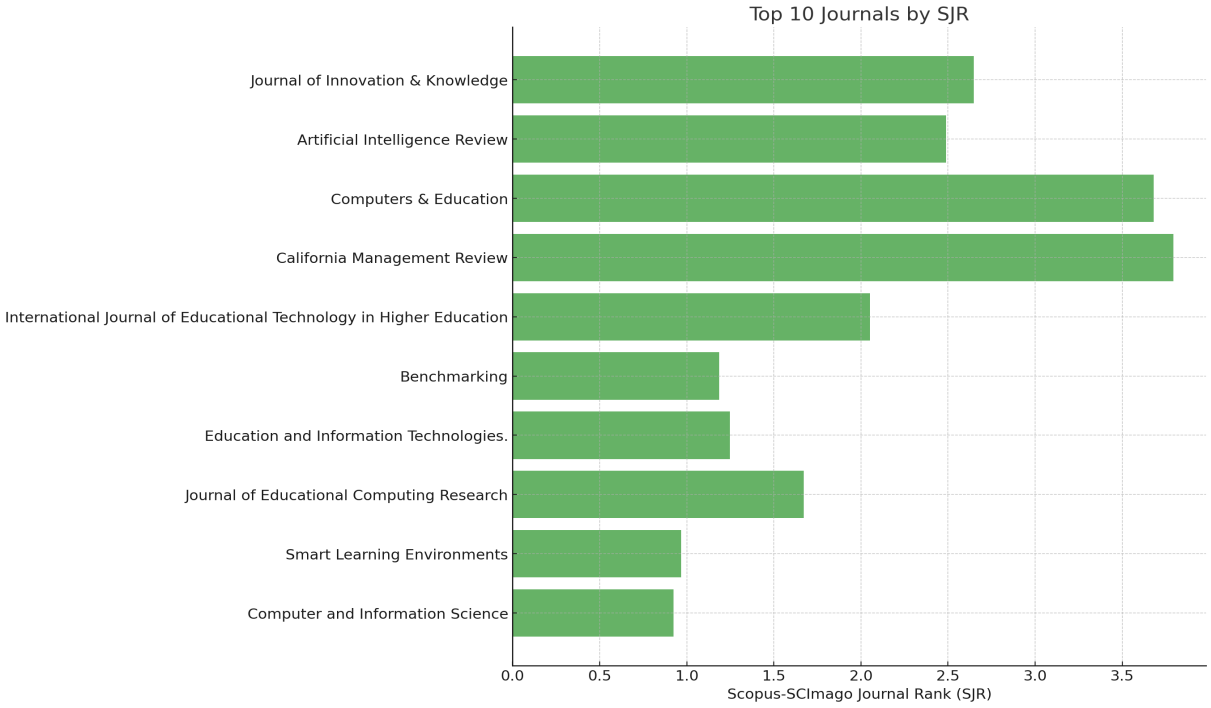


Figure 2.7 The Top 10 Journals by SJR

This comprehensive review of selected scholarly works underscores the potential and challenges of using artificial intelligence (AI) in education. AI can dramatically transform various dimensions of teaching and learning, acting as a potent force in the sector. However, alongside the many positive impacts, there are also potential pitfalls and negative impacts. This dichotomy highlights the crucial need for a robust process framework that can predict the impact of AI adoption on students' academic

performance, particularly in open and distance learning (ODL) environments. The development and refinement of such a framework cannot be overstated, as it is instrumental in harnessing the positive potential of AI while mitigating its risks. Through this balanced and thoughtful approach, the potential of AI in education can be truly unlocked. Through the analysis of various research methods, the robust capabilities of machine learning methodologies, particularly Support Vector Machines (SVM), in predicting academic outcomes have been emphasized. This illuminates the strong and growing intersection between AI and education, with machine learning emerging as a powerful tool in education research.

The reviewed studies also shed light on the crucial drivers behind the adoption of AI in distance learning contexts. They underscore its far-reaching implications on student outcomes, indicating that AI can enhance the learning experience and potentially improve educational achievement.
Furthermore, the studies underscore the potential of AI to enrich the interaction between learners and educators in digital environments. This is particularly pertinent in online and remote learning, where AI could facilitate effective teaching and learning practices. The examination of intermediary factors such as gender and geographic disparities offers a deeper understanding of the complex dynamics at play in the integration of AI in education. This nuanced understanding aids in a more comprehensive appreciation of AI's potential positive and negative impacts on the educational sector. In essence, this review emphasizes the growing significance of AI in education, its potential impacts, and the importance of ongoing research in this rapidly evolving field.

CHAPTER THREE
METHODOLOGY

3.1 Preamble

The primary objective of this study is to explore the intricate relationship between the adoption of Artificial Intelligence (AI) and students' academic performance within the realm of Open and Distance Learning (ODL) environments. Specifically, the study aims to develop a resilient predictive framework utilizing the Support Vector Machine (SVM) algorithm. This section encompasses the proposed solution, technique, research model, framework, data source, mode of data collection, sampling technique, and modelling approach.

This study seeks to construct a detailed predictive framework using the Support Vector Machine (SVM) to understand the influence of AI adoption on students' academic performance within Open and Distance Learning (ODL) settings. Central to this effort is using AI adoption determinants as predictors for academic outcomes. The SVM, recognized for its proficiency in classification and prediction based on input data, is the backbone of this project. Leveraging the SVM's capabilities, the study endeavours to produce a model pinpointing the relationship between AI adoption and students' academic results.

The methodology for this research is layered and thorough. It commences with a rigorous literature review to identify the factors affecting AI adoption and their subsequent effects on academic success in ODL contexts. This literary exploration is enriched with specific data sourced directly from ODL institutions. In the subsequent design phase, visualization tools like Visio and draw.io are utilized to craft schematic diagrams, and the Unified Modeling Language (UML) shapes the architecture of the process framework and overall research model. Data analysis is primarily executed through Python, especially within platforms such as Anaconda Navigator and Jupyter Notebook, focusing on the SVM algorithm and using libraries like Pandas, Numpy, Sklearn, matplotlib, and Imblearn. To assess the integrity and effectiveness of the developed machine learning models, they undergo evaluation using metrics like Mean Squared Error, Mean Absolute Error, and Accuracy.

3.2 Problem formulation

This study aims to develop a process framework to predict how AI adoption influences students' academic performance in ODL. AI can potentially improve learning outcomes and tailor education to individual needs, but it may also pose challenges such as loss of diversity, increased stress, and reduced autonomy. Moreover, the impact of AI adoption may vary depending on students' gender and geographical region. This study focuses on Nigeria and Canada as two contrasting cases of ODL

contexts. The study used SVM modelling as a predictive technique to create the framework. It collected and analyzed data on AI adoption and academic performance in ODL and examined how gender and geographical region moderate this relationship. The study offers valuable insights into the benefits and drawbacks of AI integration in ODL and suggests evidence-based strategies for optimizing its use. The principal goal is to enhance the quality and relevance of ODL for students across different settings.

3.3 Proposed solution, technique, model or process framework

The following section provides an in-depth outline of the proposed solution, including the technique, research model, framework, data sources, data collection methods, sampling technique, and modelling approach.

3.3.1 Machine Learning Approach

The SVM machine learning algorithm was utilized to construct the envisioned predictive model. As shown in Figure 3.1, the machine learning project workflow was adhered to in its customary stages. The machine learning life cycle was stringently observed in the present study. Through the implementation of the Python programming language, the data analysis process was executed, and the requisite algorithm was authored to create the predictive model.

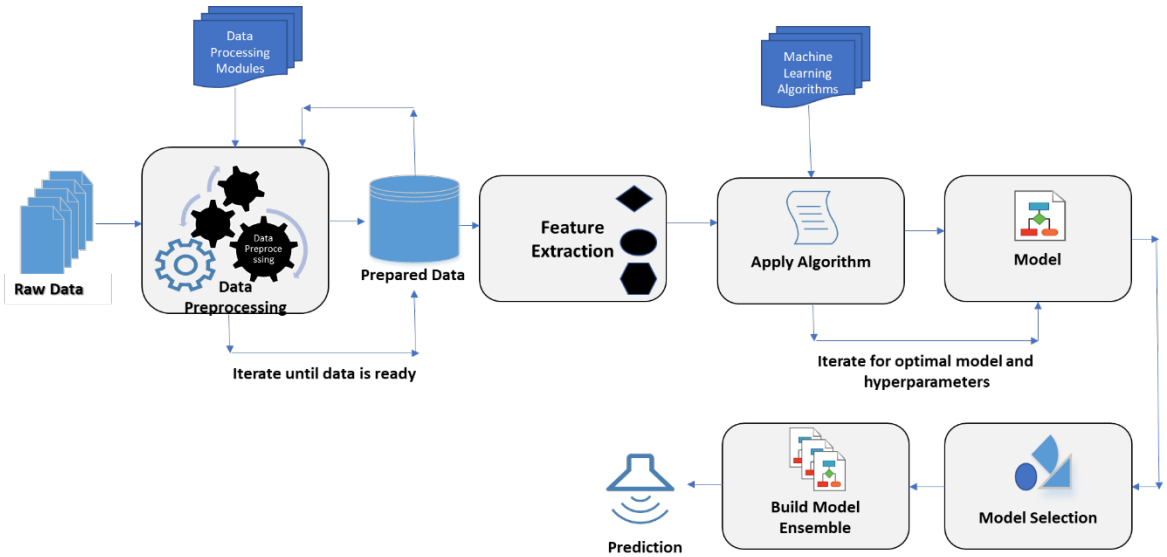


Figure 3.1 Typical Machine Learning Project Workflow

3.3.2 Proposed Process Framework

The research work utilizes a comprehensive process framework, which is detailed below. This framework is a systematic and structured approach to guide the investigation and analysis of the research objectives. The study aims to ensure coherence and effectiveness in its methodology and outcomes by following this framework. The adopted process framework encompasses several key steps that are executed sequentially to facilitate a thorough investigation.

I. **Identification of Key Factors:** This step involves identifying and selecting the key factors that influence the impact of AI adoption on students' academic performance in ODL. Relevant literature and empirical studies are reviewed to determine the critical factors significantly affecting the learning outcomes in AI-integrated ODL environments.

II. **Data Collection and Preprocessing:** In this phase, the process framework focuses on collecting relevant data related to the identified key factors. Data sources, such as student performance records, demographic information, AI usage data, and other relevant indicators, were considered. Preprocessing techniques were applied to ensure data quality and prepare the dataset for analysis.

III. **Feature Engineering and Selection:** This step involves transforming the collected data into meaningful features that can be utilized in the prediction process. Feature engineering techniques, such as data normalization, dimensionality reduction, and feature extraction, create a representative set of features for SVM modelling.

IV. **SVM Modeling and Prediction:** SVM was employed as the predictive modelling algorithm using the processed and engineered features. The SVM model was trained on historical data, leveraging its ability to analyze patterns and make predictions based on the identified key factors. The model's performance was assessed using appropriate evaluation metrics.

V. **Interpretation and Validation:** The final step of the process framework involves interpreting the results of the SVM model and validating the predictions against the actual students' academic performance in ODL. This step aims to assess the accuracy and reliability of the predictive model and gain insights into the impact of AI adoption on academic performance.

This process-driven framework (Refer to Figure 3.2) offers a systematic and organized approach to forecasting the implications of AI integration on students' academic achievements in ODL utilizing SVM. By amalgamating the theoretical underpinnings of AI, ODL, and SVM, this framework enhances the comprehension of the intricate relationships among AI integration, pivotal factors, and academic results. The framework serves as a directive for forthcoming empirical investigations and practical applications, empowering educational institutions to optimize the use of AI in ODL to elevate students' academic achievements and learning experiences. In conclusion, the methodological framework employed for this scholarly endeavour encompasses an all-encompassing approach that

entails a literature review, formulation of research inquiries, data acquisition, meticulous analysis, predictive modelling, and interpretation of results. By adhering to this framework, the research aspires to ensure methodological consistency and yield valuable insights into the correlation between AI integration and students' academic performance in ODL settings.

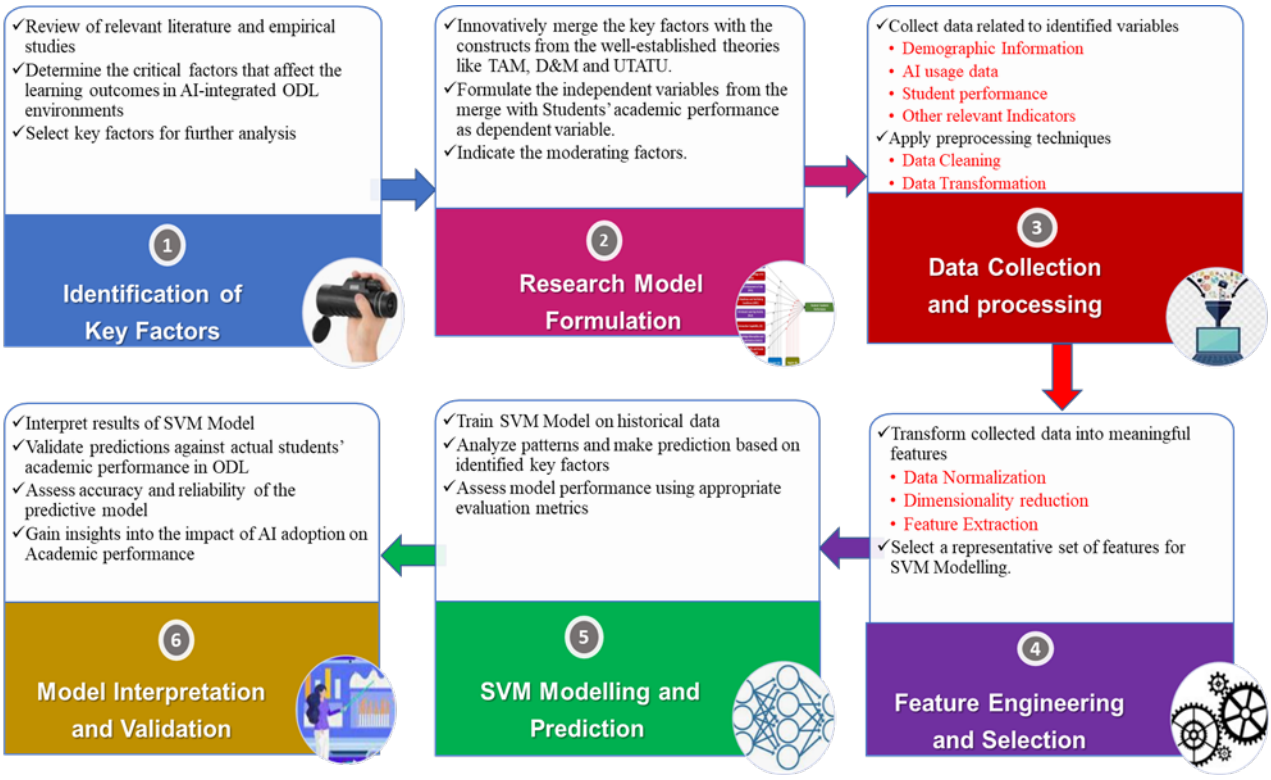


Figure 3.2 The Process Framework for Predicting the Impact of AI Adoption on Students' Academic Performance in ODL

In this diagram, each step is represented by a rectangular box, and the arrows indicate the workflow's flow. The diagram identifies key factors, followed by data collection and preprocessing. The processed data then goes through feature engineering and selection to create meaningful features for SVM modelling. The SVM model is trained on the data, and predictions are made based on the identified key factors. Finally, the results are interpreted and validated against actual academic performance data to assess the accuracy and gain insights. As described in Figure 3.2, the process Framework is independent of any predictive algorithm. The fundamental objective is to design a Process Framework that remains neutral with respect to particular machine learning algorithms. This impartiality is essential due to the ever-evolving nature of the educational environment, and associating our framework with a specific algorithm may lead to its obsolescence or diminished efficacy over time. By creating a model that is independent of any algorithm, flexibility, adaptability, and sustainability are guaranteed in its utilization.

A trio of models was intricately crafted and thoroughly compared for this doctoral study. Starting with a foundational process framework, as illustrated in Figure 3.2, the methodology evolved to integrate detailed procedural steps for each selected algorithm. This expanded and enriched layered architecture, which now includes these detailed steps, is presented in Figure 3.3. Each model, precisely crafted and aligned with research objectives, underwent a thorough comparative performance evaluation as part of the layered architecture's fifth step. This critical analysis aims to discern each model's effectiveness and accuracy, identifying its strengths and areas for improvement. This organized approach ensures the creation of robust and reliable models, aiming for academic excellence and practical applicability in understanding the research phenomena. Through strategic development and evaluation, the research aspires to unveil models that embody integrity and comprehensive analytical insights.

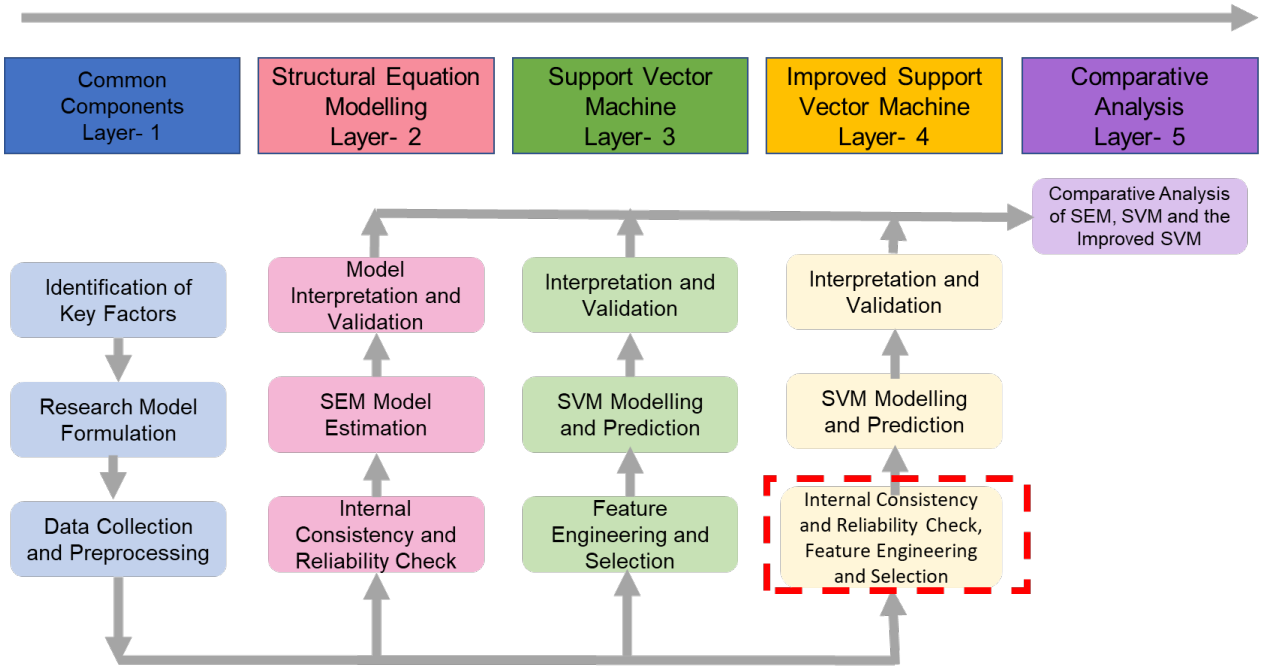


Figure 3.3 A Layered Architecture for Predicting AI Adoption on Students' Academic Performance in ODL using SEM, SVM and the improved SVM.

Figure 3.3 is the layered architecture that illustrates the distinct procedural stages of the three selected algorithms. This architecture is comprised of five layers, with the first layer encompassing three components. The second layer consists of three components. The third layer also includes three components. The fourth layer comprises three components. The fifth layer is represented by a singular component (M). Each layer has components described as follows:

Layer 1: Structural Equation Modelling (SEM) - Layer 1.

- **Identification of Key Factors:** This is the initial phase where crucial factors influencing AI adoption are identified through literature review and empirical studies.

- **Research Model Formulation:** Based on identified key factors, a research model is formulated to explore the relationships and impacts on academic performance.
- **Data Collection and Preprocessing:** This stage involves gathering data relevant to the research model and preprocessing it for analysis.

Layer 2: Structural Equation Modelling (SEM) - Layer 2

- **Internal Consistency and Reliability Check:** At this stage, the reliability of the model's constructs is assessed through methods like Cronbach's alpha.
- **SEM Model Estimation:** The SEM estimates the relationships between the identified factors and the outcomes.
- **Model Interpretation and Validation:** The final stage in SEM is where the model's findings are interpreted and validated against empirical data.

Layer 3: Support Vector Machine (SVM) - Layer 3

- **Feature Engineering and Selection:** This step focuses on selecting and engineering the most relevant features from the data for the SVM model.
- **SVM Modelling and Prediction:** An SVM model is developed to predict the outcomes based on the engineered features.
- **Interpretation and Validation:** The predictions of the SVM model are interpreted, and its performance is validated.

Layer 4: Improved Support Vector Machine (SVM) - Layer 4

- **Internal Consistency and Reliability Check, Feature Engineering and Selection:** Similar to Layer 3, but focusing on an improved SVM model that enhances the model's stability and reliability by reducing the multicollinearity among the independent variables. The reduced Variance Inflation Factor (VIF) after applying the Internal Consistency and Reliability Check confirms the reduction in multicollinearity.
- **SVM Modelling and Prediction:** This improved SVM model is employed for more accurate predictions.
- **Interpretation and Validation:** The results of the improved SVM model are interpreted and validated for their accuracy and reliability.

Layer 5: Comparative Analysis Layer - 5

- **Comparative Analysis of SEM, SVM, and the Improved SVM:** This final layer involves a comparative analysis of the results from SEM, standard SVM, and improved SVM models to determine the most effective approach for predicting the impact of AI adoption on ODL academic outcomes.

The arrows suggest the flow direction in the layered architecture for refinement and validation across the models. This ensures that each approach is rigorously evaluated and that the best model is selected

based on empirical evidence.

Based on the expanded framework in Figure 3.3, the overview of the research methodology is described in Figure 3.4 which presents an overview of the research methodology in a step-by-step, algorithmic fashion, illustrating the logical flow from one stage to the next. The process starts with defining the research objectives, which involve designing a process framework to understand AI adoption in Open and Distance Learning (ODL), developing a research model, and creating machine learning models to predict the impact of AI on student academic performance.

Following the definition of objectives, the next step is to identify key factors influencing AI adoption through a comprehensive literature review. These factors are then translated into model variables, and relationships between them are established to formulate the research model. Once the model is developed, data is collected via surveys and academic databases, and it undergoes preprocessing, including cleaning and normalization, to prepare it for analysis.

Internal consistency and reliability checks, such as Cronbach’s alpha, are applied to ensure the data's reliability. After that, feature engineering is conducted to transform the data into relevant features suitable for machine learning models, and dimensionality reduction techniques are employed if necessary. The predictive models are then developed, including Structural Equation Modeling (SEM) and Support Vector Machines (SVM), with improvements made to the SVM model to enhance predictive capabilities.

Once the models are developed, they are interpreted and validated, and the predicted outcomes are compared with actual data. A comparative analysis is performed between the SEM, SVM, and improved SVM models to determine the most effective approach. The evaluation is conducted using performance metrics such as Absolute Mean Error and Mean Squared Error to assess model accuracy and efficiency in ODL contexts.

Finally, the findings are documented, offering insights into the impact of AI adoption on academic performance within ODL systems. The validated predictive framework is then presented as a valuable tool for educational institutions to assess and optimize AI's role in enhancing learning outcomes. This clear progression of steps emphasizes the systematic approach taken in the study to ensure rigor and accuracy in the analysis.

The methodology was designed to be iterative, allowing for refinements based on findings and

validation results at each stage. This structured approach depicted in Figure 3.4 ensures a robust and comprehensive examination of AI's role in enhancing academic performance in ODL settings.

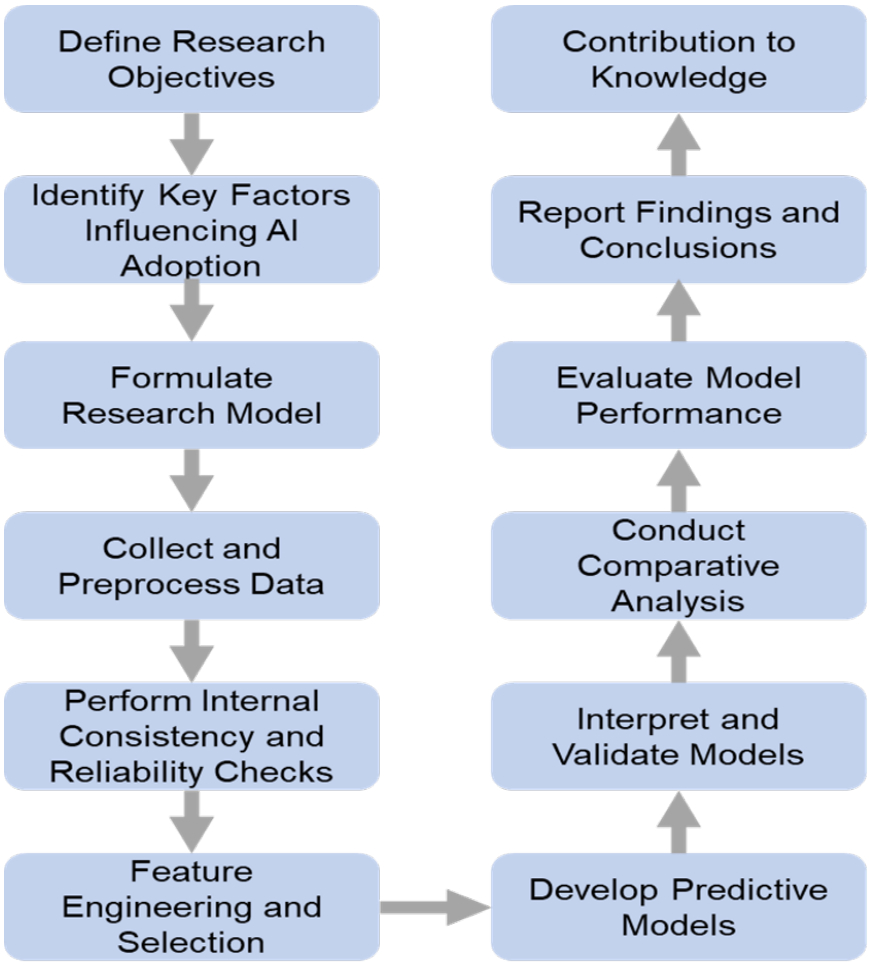


Figure 3.4 A flowchart showing the overview of the research methodology.

3.3.3 Implementation of Support Vector Machine Algorithm

Comprehending the mathematical intricacies that underlie the Support Vector Machines algorithm can unquestionably aid in understanding the implementation of the model. This understanding can provide valuable insights into selecting the most suitable model for a given problem and determining optimal values for hyper-parameters. As posited by Zhu (2021), the formulation of SVM is presented through a series of mathematical expressions delineated from Equations (1) through (13). These equations lay the foundation for constructing a hyperplane—a concept illustrated through Figures 3.5 through 3.7—effectively separating two classes in a feature space. This separation is critical for classification tasks, where the hyperplane's orientation and position, defined by vectors and margins, are optimized for the best division of classes. The mathematical particulars of Support Vector Machines are expounded below:

Consider that there are n training points, i has p features (i.e., x_i has p dimensions), and y_i is either

-1 or 1. Consider two classes of linearly separable observations. The implication is that a hyperplane can be drawn through the feature space, with all instances of one class on one side and all instances of the other class on the opposite side. (A *p-1* dimensional subspace is a hyperplane in *p* dimensions. A hyperplane is just a line in the following two-dimensional example.) A hyperplane is what is specified as:

(1)

where˜*b* is a real number and˜*w* is a *p*-vector. For ease, it is assumed that˜*w* = 1, so the distance from point *x* to the hyperplane is given by the formula *x* *˜*w* +˜*b*.

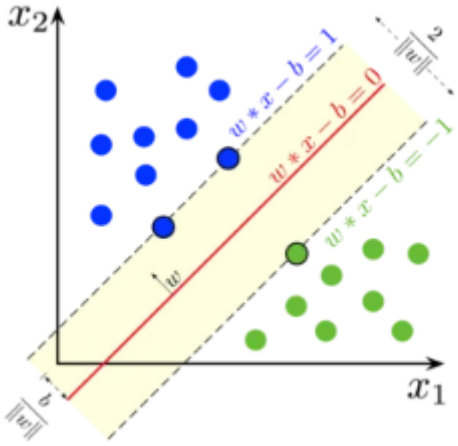


Figure 3.5 Key Concepts of SVM (Source: Zhu (2021))

Thus, the condition that the hyperplane divides the classes can be met by labelling the classes with *y* = +1/-1:

$$y_i\left(x_i \tilde{} w + \tilde{} b\right) \geq 0 \tag{2}$$

The Maximal Margin Classifier selects the plane that yields the largest margin M between the two classes and determines the best hyperplane.

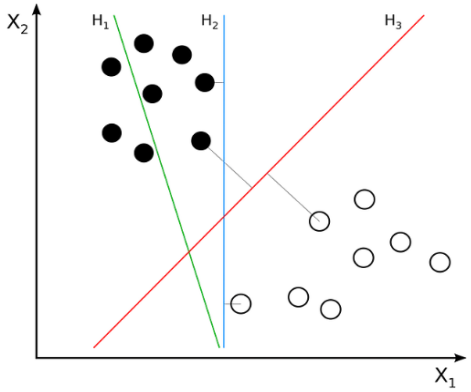


Figure 3.6 How to choose the best hyperplane (Source: Zhu (2021))

H_1 does not distinguish between the two classes in the previous graph; for H_2 and H_3, H_3 is chosen because H_3 has a larger margin. Given the constraints, mathematically, $\tilde{b}$ and $\tilde{w}$ are selected to maximize M:

$$y_i\left(x_i \tilde{.} w + \tilde{b}\right) \geq M \tag{3}$$

Defining w = $\tilde{w}$ / M and b = $\tilde{b}$ / M, this can be rewritten as:

$$y_i\left(x_i . w + b\right) \geq 1 \tag{4}$$

and

$$\|\tilde{w}\| = 1, \|\tilde{w}\| = \frac{1}{M} \tag{5}$$

Support vectors present a significant challenge in classification since they are the data points closest to the separating hyperplane. Their elimination would change the positioning of the dividing hyperplane, which is exclusively influenced by the support vectors through a weight-generating optimization algorithm. The optimization algorithm for generating the weights operates so that only the support vectors are accountable for determining both the weights and the boundary. Mathematically, support vectors can be defined as those points which are in closest proximity to the decision boundary and are defined as:

$$x_i * w + b = -1 \quad \textit{for negative class} \tag{6}$$

$$x_i * w + b = 1 \quad \textit{for positive class} \tag{7}$$

The hard-margin support vector machine (SVM) is a rigid method that imposes strict constraints on the support vectors that cross the hyperplane. It is designed to disallow any support vectors from being incorrectly classified. The optimization problem faced by the hard-margin SVM aims to maximise the hyperplane's margin.

$$\begin{gathered} \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 \\ \text{subject to } y_i(\boldsymbol{x_i} \cdot \boldsymbol{w} + b) \geq 1 \\ \text{for } i = 1, \ldots, n \end{gathered} \tag{8}$$

Soft-margin support vector machines (SVMs) are commonly used when dealing with non-linearly separable classes. The reason for such difficulty may be attributed to the absence of a clear class boundary or the presence of a non-linear boundary. To address this issue, SVMs employ slack variables, which permit a few points to cross or deviate from the margin. This can be observed in the

accompanying graph. Hyper-parameter C controls the extent to which the slack variables are allowed to influence the SVM's decision boundary.

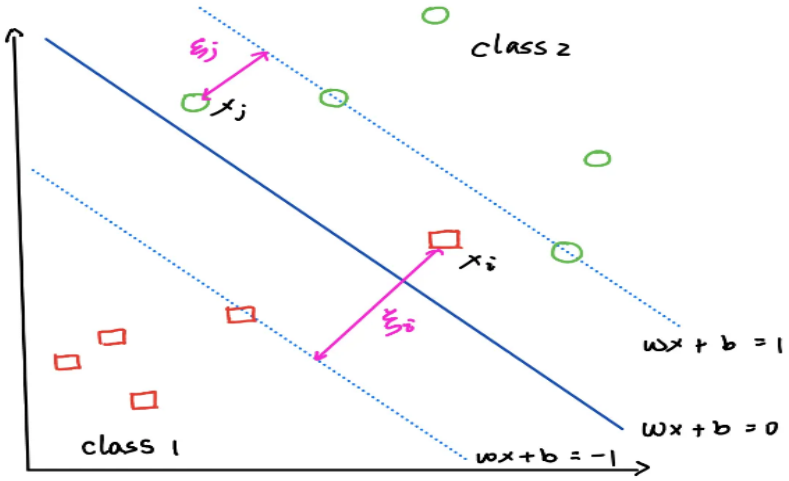


Figure 3.7 Soft-margin SVM and the hyper-parameter C (Source: Zhu (2021))

The soft-margin support vector machine aims to optimize the objective function by minimizing slacks and maximizii

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\frac{1}{n}\sum_i \xi_i \tag{9}$$

$$\text{subject to } \begin{cases} y_i(\boldsymbol{x}\cdot\boldsymbol{w}+b) \geq (1-\xi_i) & \text{for } i = 1,\ldots,n \\ \xi_i \geq 0 & \text{for } i = 1,\ldots,n \end{cases}$$

The primal problem in optimization involves a constant C that represents the "cost" of slack. A smaller value of C is preferable when allowing more points into the margin is efficient, as it achieves a more significant margin. By increasing the number of support vectors, SVM reduces its variance, making the model more generalized. Therefore, decreasing C increases the number of support vectors and reduces overfitting. With Lagrange multipliers:

$$\underbrace{\alpha_i \geq 0 \text{ and } \mu_i \geq 0}_{\text{two constraints}} \tag{10}$$

The problem of constrained optimization can be rephrased as a primal Lagrangian function:

$$\min_{\boldsymbol{w},b,\xi}\max_{\alpha,\mu}\left[\frac{1}{2}\|\boldsymbol{w}\|^2 + C\frac{1}{n}\sum_i \xi_i - \sum_i \alpha_i\left[y_i(\boldsymbol{x_i}\cdot\boldsymbol{w}+b)-(1-\xi_i)\right] - \sum_i \mu_i\xi_i\right] \tag{11}$$

The dual Lagrangian formulation involves maximizing over the multipliers based on previously obtained relations for w and b rather than minimizing over w and b, subject to constraints.

$$\max_{\alpha} \left[\sum_{i} \alpha_i - \frac{1}{2} \sum_{i,i'} \alpha_i \alpha_{i'} y_i y_{i'} \boldsymbol{x_i} \cdot \boldsymbol{x_{i'}} \right] \tag{12}$$

$$\text{subject to } \begin{cases} 0 = \sum_i \alpha_i y_i \\ 0 \leq \alpha_i \leq C \quad \text{for } i = 1, \ldots, n \end{cases}$$

The task of optimizing a quadratic programming problem can be effectively tackled through the utilization of the Sequential Minimization Optimization methodology, and once optimized, the coefficients can be easily determined:

$$\boldsymbol{w} = \sum_{i} \alpha_i y_i \boldsymbol{x_i} \tag{13}$$

Walking through the mathematical underpinnings of Support Vector Machines is crucial in comprehending its implementation, as it guides the selection of the appropriate model for specific inquiries in this study and the determination of the optimal values for hyper-parameters.

In the context of Support Vector Machines (SVM) with a linear kernel, interpreting the impact of each predictor on the target variable can be more straightforward compared to non-linear kernels. When the predictors (features) are standardized before fitting the SVM model, each variable has been scaled to have a mean of zero and a standard deviation of one. This standardization allows for a more direct comparison of the coefficients' magnitudes in terms of their relative importance or impact on the target variable. However, it is important to note that SVMs do not provide coefficients like linear regression, but the weights (coefficients) in a linear SVM can still offer insights.

Here is how to interpret the impact of each predictor in a linear kernel SVM:

Understanding the Weights of a Linear SVM

For a linear SVM, the decision function is given by Equation (14) as:

$$f(x) = w^T x + b \tag{14}$$

- $\boldsymbol{w}$ is the weight vector, where each weight corresponds to a feature (predictor).
- $\boldsymbol{x}$ is the feature vector.
- $\boldsymbol{b}$ is the bias term.

The weight vector $\boldsymbol{w}$ holds the key to understanding the impact of each predictor on the target variable. Each weight in $\boldsymbol{w}$ corresponds to a feature, and its magnitude indicates the importance of that feature in determining the margin between the classes.

Interpreting the Weights:

I. Magnitude: The magnitude of each weight (ignoring the sign) indicates the relative importance of that feature in classifying the target variable. Larger magnitudes mean that

the feature has a more significant impact on the decision boundary. Since the variables were standardized, these magnitudes can be directly compared to assess the more important features.

II. Sign: The sign of each weight indicates the direction of its impact. A positive weight suggests that higher values of that feature push the prediction towards one class, while a negative weight suggests that higher values push the prediction towards the other.

The steps to Interpret the Model are as follows:

I. Extract Weights: After fitting the linear SVM model, extract the weight vector w. This is typically accessible directly from the model object in most machine learning libraries like scikit-learn (e.g., ***model. coef_***).

II. Examine the Weights: Look at the magnitude and sign of each weight to understand each predictor's relative importance and direction of influence.

III. Report: For reporting, features and their corresponding weights can be listed, highlighting which features are most influential in the model and in which direction they influence the target variable.

3.3.4 Test for Multicollinearity

The Variance Inflation Factor (VIF) serves as a statistical instrument employed to identify multicollinearity among predictors within a regression framework. VIF assesses the extent to which the variance of an estimated regression coefficient is augmented as a result of multicollinearity. It quantifies the escalation in the variances of the regression parameter estimations attributable to collinear relationships among the predictors. A commonly accepted guideline posits that a VIF value surpassing 10 signifies substantial multicollinearity (Kim, 2019). The computation of VIF is delineated in Equation (15) for each predictor variable as follows:

$$VIF_i = \frac{1}{1 - R_i^2} \quad (15)$$

Where R_i^2 is the coefficient of determination derived from the regression of predictor i on all the other predictors. An elevated VIF suggests that the predictor exhibits a strong correlation with other predictors, thereby complicating the evaluation of the distinct contribution of each predictor to the variation observed in the response variable (Salmerón, García, & García, 2020).

In this study, reducing multicollinearity among the independent variables prior to applying the machine learning algorithms proved to be highly beneficial. By ensuring that the predictors were not

highly correlated, the interpretability of the model was successfully improved, making it easier to understand the individual impact of each variable on the dependent variable. This reduction in multicollinearity also enhanced the stability of the model, preventing the coefficients from becoming overly sensitive to small changes in the data, thereby increasing the reliability of our results. Additionally, minimizing multicollinearity contributed to more accurate predictions by allowing the model to discern the true relationships between predictors and outcomes. It also helped in reducing the risk of overfitting, ensuring that the model did not rely on redundant information and thus performed better on new, unseen data. Furthermore, addressing multicollinearity streamlined the feature selection process, simplifying the identification of the most relevant predictors for the model. Overall, these efforts significantly improved the robustness and effectiveness of the predictive models developed in this research.

3.3.5 Study Area and Target Population

The study focuses on Open and Distance Learning (ODL) environments in two distinct geographical regions: Nigeria and Canada. These regions were chosen to represent different educational contexts and cultural backgrounds. The focus demographic encompasses students currently registered within ODL curricula in both nations., encompassing diverse age groups, academic disciplines, and educational levels.

3.3.6 Source of Data

The primary data source for this study is collected from the respective ODL institutions in Nigeria and Canada. Institutional collaboration and partnerships are established to gain access to the necessary data. Ethical considerations and institutional protocols are followed to ensure the confidentiality and privacy of the participants' information. The study employed a purposive sampling technique to select participants from ODL institutions. The sample included diverse students from various disciplines and educational levels. The goal is to comprehensively understand the impact of AI adoption on academic performance in different contexts. A large and diverse dataset was acquired by leveraging the power of quantitative survey methodology, enabling comprehensive analysis and reliable results.

3.3.7 Methods of data collection

Data were collected through surveys. The surveys were designed to gather relevant information about students' demographics, AI adoption in ODL, academic performance indicators, and the perceived impact of AI on their learning outcomes. The educational records provide objective measures of academic performance, such as grades, completion rates, and assessment scores. The survey questionnaires were distributed online using established data collection and management platforms.

The Cumulative Grade Point Average was obtained from the respective students, following the necessary data protection protocols and permissions.

3.3.8 Sample Size Determination

This section focuses on the rationale behind the determination of the sample size. The sample size is a critical aspect of research design that significantly impacts the reliability and validity of the study's findings. An adequately sized sample ensures that the study results are generalizable to the broader population while also providing sufficient power to detect meaningful effects or differences when they exist.

The importance of selecting an appropriate sample size cannot be overstated. A sample size that is too small may lead to a lack of statistical power, increasing the risk of Type II errors (failing to detect an existing effect). Conversely, a sample size that is too large may result in wasted resources and potentially increase the risk of Type I errors (detecting an effect that does not exist due to random chance). Thus, determining the optimal sample size is crucial for balancing these risks while ensuring the efficient use of resources (Shen et al., 2014).

This study's sample size determination was guided by several critical factors, including the study design, the expected effect size, the desired power level, and the significance level (alpha). The effect size refers to the magnitude of the difference or relationship the study aims to detect, which could be based on previous studies or theoretical considerations. The power of the study, typically set at 80% or higher, indicates the probability of correctly rejecting the null hypothesis when it is false. The significance level, often set at 0.05, defines the threshold for determining statistical significance (Albers & Lakens, 2018).

The formula for estimating sample size in quantitative studies was employed, taking into account the aforementioned factors to calculate the sample size. For instance, in comparing two means, the sample size for each group can be calculated using the formula as depicted in Equation (16):

$$n = \left(\frac{Z_{\frac{\alpha}{2}} + Z_{\beta}}{\delta}\right)^2 \sigma 2 \quad (16)$$

Where n is the sample size per group, $Z_{\alpha/2}$ is the critical value of the normal distribution at $\alpha/2$ (for a two-tailed test), Z_{β} is the critical value of the normal distribution at the desired power (β), δ is the expected effect size, and σ^2 is the variance within the population. For analyses involving correlations or regressions, sample size determination was informed by similar considerations but tailored to the specific statistical tests used. These calculations were guided by tools such as G*Power and

consultation with statistical texts (Faul et al., 2019).

Based on a preliminary literature review and the expected effect size derived from similar studies, this study's desired sample size was 790, assuming a power of 90% and a significance level of 0.05. This sample size is deemed sufficient to detect the expected effects within the constraints of the study's design and objectives. Determining the sample size was a critical step in the research design, ensuring that the study is adequately powered to detect meaningful differences or relationships while considering practical limitations and ethical considerations. The calculated sample size supports the study's goals of producing reliable, valid, and generalizable findings that contribute meaningfully to the existing knowledge of AI in education. This meticulous approach to sample size determination underscores the rigour and thoughtfulness of the research methodology, setting a solid foundation for the subsequent data collection and analysis phases.

3.3.9 Methods of Analysis

The data gathered undergoes a rigorous analysis utilizing statistical techniques and machine learning algorithms, explicitly focusing on Support Vector Machine (SVM) modelling. SVM was employed to predict the impact of AI adoption on students' academic performance, considering the moderating factors of gender and geographical region.

Descriptive statistics is utilized to examine the demographic characteristics of the participants and the level of AI adoption in ODL. Inferential statistics, which encompass correlation analysis and structural equation modelling, were conducted to explore the relationships between AI adoption, academic performance, and moderating factors.

The SVM algorithm is employed to develop a predictive model that can anticipate the impact of AI adoption on students' academic performance. A model is developed to forecast students' academic performance based on AI adoption factors. The collected data are used to train and validate the model, and its performance is assessed using appropriate evaluation metrics. An SVM (Support Vector Machine) model is considered alongside Structural Equation Modeling (SEM) for several strategic reasons in this research:

I. **Different Focus:** SEM establishes and validates relationships between observed and latent variables. It is excellent for hypothesis testing and model fitting based on observed data. On the other hand, SVM is a machine-learning model primarily used for classification and regression. It focuses on predictive accuracy and generalization to new, unseen data.

II. **Predictive Accuracy:** SVM is renowned for its high predictive accuracy and ability to handle high-dimensional data spaces effectively. It can robustly manage non-linear relationships and

interactions between variables, enhancing the model's predictive performance.

III. **Handling Non-linearity:** SVM can effectively manage non-linear relationships in the data through kernel functions, enabling the model to capture complex relationships and interactions, which SEM may not easily handle.

IV. **Robustness:** SVM is less sensitive to specification errors and is robust in noisy data. It is more focused on minimizing prediction errors, making it a robust tool in scenarios where prediction is key.

V. **Generalization:** SVM emphasizes the model's ability to generalize to new data, ensuring that the findings fit the sample data and apply to broader contexts.

VI. **Complementary Approach:** Using SVM alongside SEM allows for a complementary approach where SEM can help understand the underlying relationships and pathways. At the same time, SVM can enhance the predictive aspect, providing a well-rounded analysis.

VII. **Objective Alignment:** The research aims to develop a predictive framework. SVM aligns with this goal by offering a tool specifically designed for forecasting and prediction, complementing the insights derived from SEM.

By incorporating SVM alongside SEM, the research can leverage the strengths of both methodologies, combining SEM's capability in model fitting and hypothesis testing with SVM's robust predictive capabilities, ensuring a comprehensive and robust analysis aligned with the research objectives.

3.4 Tools used in the implementation

This section outlines the digital and analytical tools essential for the implementation of research. This section lists each tool used, comprehensively describing the tools' functions and their specific roles in the study context, as detailed in Table 3.1. The descriptions aim to elucidate the tools' contributions to data collection, analysis, or other research processes they facilitated. This allows for methodology transparency and offers readers insights into the practical aspects of the research's technical execution.

Table 3.1 Tools used in the implementation

S/N	Tool	Description of the Tool	How it is Used in the Research
1	Comprehensive Literature Review	Method for gathering existing knowledge	Identify factors affecting AI adoption and their effects on academic success in ODL contexts.
2	Visio, draw.io & Ludichart	Visualization tools for diagrams and charts	Crafting schematic diagrams during the design phase.
3	Unified Modeling Language (UML)	Language for specifying, visualizing, constructing, and documenting software systems	They are shaping the architecture of the process framework and overall research model.
4	Python	Object-oriented High-level programming language	The primary language for data analysis, especially within platforms like Anaconda Navigator and Jupyter Notebook
5	SVM Algorithm	A machine learning algorithm for classification and regression	Focus on prediction in the developed machine learning model.
6	Pandas, Numpy, Sklearn, matplotlib, and Imblearn	Libraries in Python for data analysis and visualization	Used with Python for data analysis, processing, visualization, and machine learning tasks.
7	Evaluation Metrics (Mean Squared Error, Mean Absolute Error, and Accuracy)	Evaluation metrics for machine learning models	These are the metrics to assess the integrity and effectiveness of the developed machine learning models.
8	Grammarly	Online writing and grammar checking tool	This was used to manage and check the quality of written content and ensure grammatical accuracy.
9	Citation Generator	Tool for generating citations in various formats	Managing citations throughout the research process.
10	Search Engine	Digital tools for finding specific information on the World Wide Web	Conducting additional background checks, referencing, and verification of sources
11	Questionnaire/Google Form	Tools to collect data from respondents, designed with structured queries	Gathering primary data, collecting responses related to the study's focus, and obtaining participant feedback or insights.
12	PowerPoint Deck	Tool for presenting the work to the supervisors, committees and International conferences	The work was laid out in PowerPoint slides and presented to the supervisors, committees and International conferences.
13	Smart PLS	Statistical analysis tool for carrying out data analysis using Structural Equation Modelling	The collected data was fed into Smart PLS and analysed using the defined research and structural equation models.

3.5 Approach and Techniques for the Proposed Solution

A detailed and structured approach is paramount to navigate the complexities and achieve this investigation's aims. This section delves into the methodologies and techniques to ensure the study's effectiveness. It narrates the meticulous planning and precise execution that underpin the research, from the initial conceptual framework to the refinement of specific algorithms. The narrative

elucidates the progression from the initial concept to practical implementation, discussing the framework's design, the crafting of the model, the algorithm's development, and the formulation of the operational scheme. By dissecting these elements, the section offers a lucid exposition of the methodologies and strategies utilized, highlighting the exacting methods undertaken to realize the research's proposed solutions.

3.5.1 Design of Framework

To better elucidate the research objectives and the practical steps towards their realization, let us explore the intricacies involved in the 'Design of the framework' and how it serves to achieve these objectives:

Objective 1: Design a process framework incorporating the factors identified from the requirements to enhance understanding of AI adoption in Open Distance Learning (ODL).

Activity 1.1: A thorough systematic review of literature pertaining to AI integration in Online Distance Learning (ODL) was conducted. The focus was on identifying the determinants that propel the acceptance of artificial intelligence in such settings, examining the effect on students' academic performance, and understanding gender and geographical variances in AI adoption. This review draws from various sources, including electronic databases, academic journals, conference proceedings, and other pertinent materials. Studies aligning with the research objectives were selected using established inclusion and exclusion criteria. Data extracted from these studies were rigorously analyzed, providing a comprehensive perspective on the current state of AI adoption in ODL.

Activity 1.2: The inputs to this activity are the outputs from activities 1.1, 3.1 and 3.2. This activity involves designing the questionnaire and initiating and circulating an online questionnaire among ODL students. This online questionnaire captured data regarding AI adoption influencers, academic performance metrics, and essential demographic details. It employed a combination of text mining techniques and quantitative survey methodology via cluster sampling to collect data from student populations regarding their use of AI-based applications in the classroom. Cluster sampling was utilised to randomly select schools from each state, forming clusters and administering an online data collection form via Microsoft Forms. The instrument consisted of demographic data and data on factors influencing the developed conceptual model.

Activity 1.3: Develop a comprehensive process model that weaves in the factors discerned from the requirements elicitation phase, aiming to amplify insights into AI adoption within ODL settings. This model chronologically mapped out the key stages:

I. **Identification Phase:** Pinpoint the pivotal factors driving AI adoption in the ODL environment, emphasizing their interplay with student academic performance.

II. **Design and Validation Phase:** Design the research model and the process framework considering the dynamics of these factors and validate its representational accuracy.
III. **Implementation Phase:** Deploy machine learning techniques, capitalizing on gathered datasets, to predict how AI adoption determinants influence academic results.
IV. **Evaluation Phase:** Scrutinize the efficacy of the implemented system and machine learning models, ensuring they resonate with the primary aim of understanding AI's role in shaping academic outcomes in ODL.

This activity integrates the essence of the other objectives, particularly emphasizing the progression from requirements elicitation to evaluation, ensuring a holistic understanding of AI adoption's nuances in the ODL setting. All the other activities are inputs to activity 2.1.

Objective 2: Design a research model comprising the factors of AI adoption and student academic performance in ODL.

Activity 2.1: Construct a conceptual model using the core constructs of renowned theories - TAM, D&M, and UTAUT combined with specific factors inherent to the ODL context, refining them into eight primary independent variables. These variables are AI Alignment and Relevance (AAR), Comparative Advantage of AI (CAAI), Ease and Enjoyment of Use (EEU), AI Readiness and Facilitating Conditions (ARFC), AI-induced Learning Anxiety (AILA), Interactive Capability (IC), Knowledge Absorption and User Satisfaction (KAUS), Systems Quality and Social Influence (SQSI). These variables play a crucial role in shaping the adoption and application of AI technologies within the ODL framework, where students' academic performance is the main dependent outcome. Based on these, related hypotheses were established.

Activity 2.2: Furthermore, to provide a more comprehensive understanding, gender (G) and geographical location/region (R) were incorporated as moderating factors. This shows how gender and regional differences influence AI adoption within the ODL setting. The input to this activity is the output from activity 2.1.

Objective 3: Develop machine learning models to predict the impact of the identified factors of AI adoption on student academic performance.

Activity 3.1: Preprocess the received data from activity 2.3 to ensure its readiness for further scrutiny.

Activity 3.2: Utilized advanced algorithms, such as SVM, to construct a predictive model that correlates the determinants of AI adoption with academic outcomes. This model primarily focuses on understanding the relationship between AI adoption and student performance. Moreover, potential enhancements to the SVM algorithm were explored to improve its accuracy.

Activity 3.3: For more profound validation of the SVM outcomes achieved in Activity 4.1, apply structural equation modelling (SEM). This approach encompasses confirmatory factor analysis and validation, ensuring the alignment of SEM findings with the insights derived from machine learning. The machine learning model's outcomes are additionally validated through SEM analysis, thereby enhancing the credibility of the findings.

Activity 3.4: Analyzed the collected data to pinpoint the factors that drive AI adoption in ODL. This involved understanding the relationships between identified determinants and recognizing disparities in adoption based on gender and regional nuances. Subsequently, gauge the influence of these AI adoption drivers on students' academic performance, especially with respect to gender and regional variations.

Objective 4: Evaluated the machine learning models of AI adoption and student academic performance to establish the level of accuracy.

Activity 4.1: Evaluated the predictive model's performance by assessing various metrics, such as mean absolute error, Mean squared error, etc. The model's effectiveness was validated in predicting students' academic performance based on AI adoption.

Process Map: Understanding AI Adoption in ODL

An overview of the entire process framework is given in Figure 3.2, and more details are provided in section 3.4. The more detailed description of the process map for research as the whole is further elaborated below, detailing the description, inputs and outputs for each activity as follows:

Objective 1: Design a process model for AI adoption in ODL.

Activity 1.1: Systematic Review of AI in ODL Literature

- **Inputs:** Existing ODL literature, electronic databases, academic journals, conference proceedings, and other sources were thoroughly reviewed.
- **Outputs:** Identified the factors driving AI adoption, understanding of effects on students' academic performance, and understanding of gender and geographical variances.

Activity 1.2: Questionnaire Distribution and Data Collection

- **Inputs:** Findings from Activity 1.1, 3.1, and 3.2.
- **Outputs:** Data on AI adoption influencers, academic performance metrics, and essential demographic details were collected.

Activity 1.3: Process Model Development

- **Inputs:** Outputs from all other activities are the inputs.

- **Outputs:** A process model detailing the Identification, Design and validation, Implementation, and Evaluation phases.

Objective 2: Design a research model for AI adoption and academic performance in ODL.

Activity 2.1**:** Conceptual Model Creation

- **Inputs:** Established theories (TAM, D&M, UTAUT), ODL-specific factors.
- **Outputs:** Conceptual model with eight primary independent variables and their hypotheses.

Activity 2.2: Incorporation of Gender and Geographical Differences

- **Inputs:** Output from Activity 1.1.
- **Outputs:** Enhanced understanding of gender and geographical influence on AI adoption in ODL.

Objective 3: Develop predictive models for AI adoption's impact on academic performance.

Activity 3.1: Data Preprocessing

- **Inputs:** Data from Activity 2.3.
- **Outputs:** Cleaned and prepared data ready for analysis.

Activity 3.2: SVM Model Creation and Refinement

- **Inputs:** Processed data.
- **Outputs:** Predictive model, potential SVM enhancements.

Activity 3.3: Validation with SEM

- **Inputs:** SVM outcomes from Activity 4.1.
- **Outputs:** Validated findings through SEM.

Activity 3.4: Analyzing Collected Data

- **Inputs:** Collected data from previous activities.
- **Outputs:** Factors driving AI adoption, relationships between determinants, understanding of adoption disparities.

Objective 4: Evaluation of machine learning models.

Activity 4.1: Model Performance Evaluation

- **Inputs:** Predictive model's outcomes.
- **Outputs:** Evaluation metrics (accuracy, Error costs, etc.), validation of model's predictive power.

Figure 3.8. shows the process map for the designed Process Framework. The findings from the

research were synthesized, and the factors influencing AI adoption and their impact on academic performance in ODL were concluded. Based on the results, provide recommendations for designing and implementing effective AI-based interventions to enhance academic performance in ODL systems. By executing these activities systematically and cohesively, the research objectives outlined above can be achieved effectively. The resulting findings contribute to advancing knowledge in AI adoption in Online Distance Learning (ODL) and facilitate designing and implementing effective AI-based interventions to enhance academic performance in ODL systems.

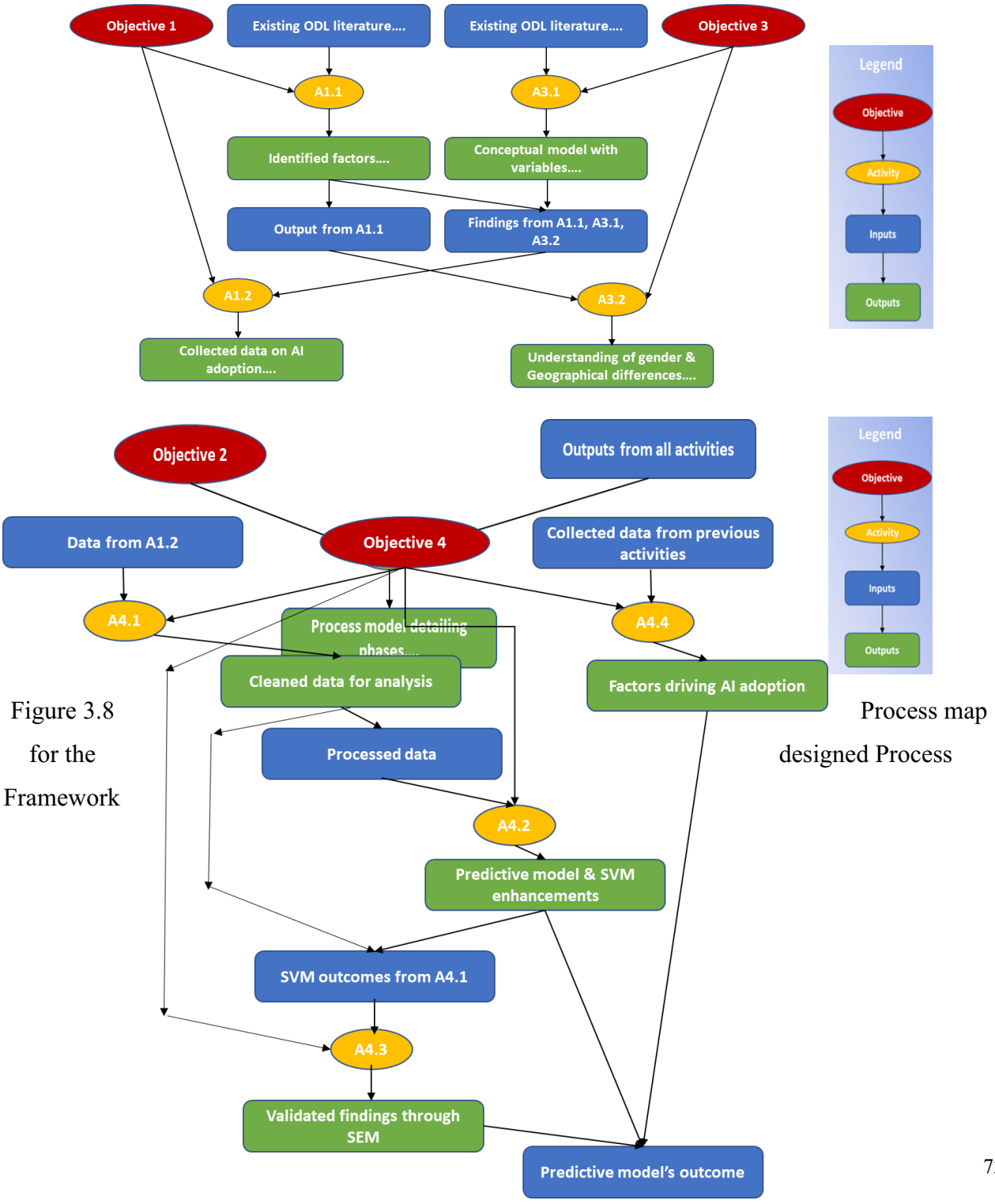
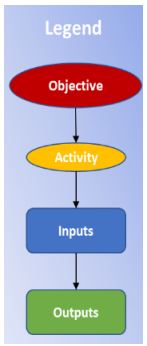


Figure 3.8 Process map for the designed Process Framework



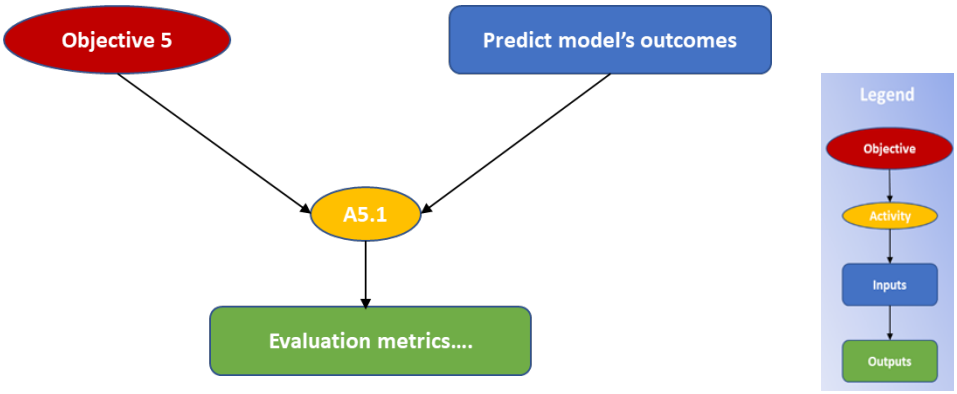


Figure 3.8 Process map for the designed Process Framework (Contd.)

This study aims to investigate the intricate relationship between AI adoption and students' academic performance in ODL settings. A predictive framework was developed by employing the SVM algorithm and integrating key constructs from established frameworks. Through comprehensive data collection, cluster sampling, and a machine learning modelling approach, this study seeks to provide valuable insights and inform effective interventions to enhance academic performance through AI adoption in ODL systems.

3.5.2 Formulation of model

This study combined the core constructs of renowned theories - TAM, D&M, and UTAUT- with

specific factors inherent to the ODL context, refining them into eight primary independent variables. These variables, depicted in Figure 3.9, directly influence the implementation and utilisation of modern AI technologies in ODL settings, with students' academic performance as the dependent variable. The influence of these primary factors is further moderated by Gender(G) and Geographical Location/Region(R):

I. **AI Alignment and Relevance (AAR):** Measures AI's fit with student and institutional needs, integrating Institutional Alignment, Attitude toward Technology, and facets of Perceived Usefulness (Charness & Boot, 2016; Sabeh et al., 2021).
II. **Comparative Advantage of AI (CAAI):** Assesses the benefits of AI versus traditional methods, integrating Comparative Advantage and aspects of Perceived Usefulness (Yakubu & Dasuki, 2018).
III. **Ease and Enjoyment of Use (EEU):** Gauges AI use's simplicity and pleasure, blending Perceived Ease of Use and Perceived Enjoyment (Sabeh et al., 2021).
IV. **AI Readiness and Facilitating Conditions (ARFC):** Evaluates readiness for AI adoption and existing supportive conditions (Sabeh et al., 2021).
V. **AI-induced Learning Anxiety (AILA):** Determines the stress linked to AI-based learning.
VI. **Interactive Capability (IC):** It assesses preparedness for and enhancements in AI-facilitated online interactions (Charness & Boot, 2016; Sabeh et al., 2021).
VII. **Knowledge Absorption and User Satisfaction (KAUS):** Examines AI's impact on knowledge uptake and overall user contentment (Yakubu & Dasuki, 2018).
VIII. **Systems Quality and Social Influence (SQSI):** Evaluates AI system quality and the role of societal factors in its adoption (Sabeh et al., 2021; Yakubu & Dasuki, 2018).

This study presents a model blending constructs from prominent theories like the Technology Acceptance Model (TAM) - focusing on technology's ease of use and perceived usefulness (Charness & Boot, 2016); DeLone & McLean's Information Systems Success Model (D&M) - emphasizing system quality and user satisfaction (Sabeh et al., 2021); and the Unified Theory of Acceptance and Use of Technology (UTAUT) - assessing factors influencing technology acceptance, such as performance expectancy, effort expectancy, social influence, and facilitating conditions (Yakubu & Dasuki, 2018). Table 3.2 displays the research model's independent variables, which are formed by merging theoretical constructs with ODL-specific elements. These combined constructs constitute the model's independent variables.

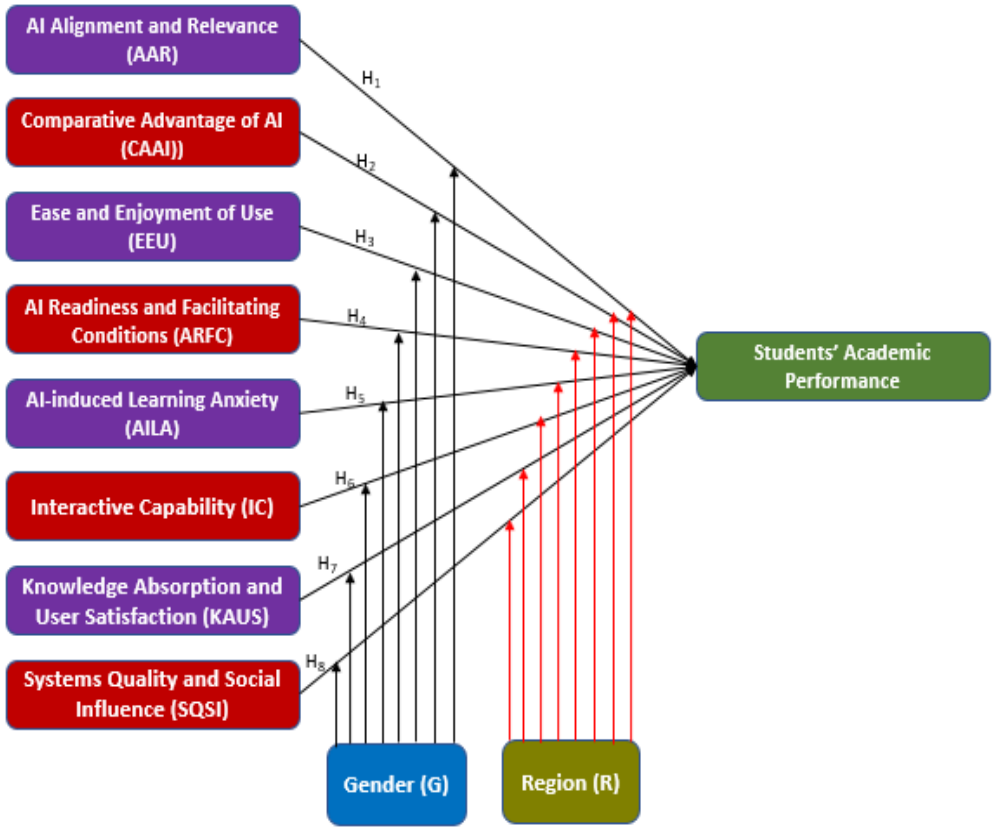


Figure 3.9 Research Model

The following Hypothesis H1, H2 to H8 were tested in this study:

I. H1: AI Alignment and Relevance (AAR) significantly impacts Students' academic performance prediction.

II. H2: Comparative Advantage of AI (CAAI) significantly impacts Students' academic performance prediction.

III. H3: Ease and Enjoyment of Use (EEU) significantly impacts Students' academic performance prediction.

IV. H4: AI Readiness and Facilitating Conditions (ARFC) significantly impacts Students' academic performance prediction.

V. H5: AI-induced Learning Anxiety (AILA) significantly impacts Students' academic performance prediction.

VI. H6: Interactive Capability (IC) significantly impacts Students' academic performance prediction.

VII. H7: Knowledge Absorption and User Satisfaction (KAUS) significantly impact Students' academic performance prediction.

VIII. H8: Systems Quality and Social Influence (SQSI) significantly impacts Students' academic performance prediction.

Table 3.2 Constructs of the newly formulated research model

S/N	Research model variables	Established Theories Constructs	Elements Unique to ODL
I	AI Alignment and Relevance (AAR) ***(Measures AI's fit with student and institutional needs)***	Attitude toward Technology (TAM) Facets of Perceived Usefulness (UTAUT)	Institutional Alignment
II	Comparative Advantage of AI (CAAI) ***(Assesses the benefits of AI versus traditional methods)***	Aspects of Perceived Usefulness (UTAUT)	Comparative Advantage
III	Ease and Enjoyment of Use (EEU) ***(Gauges AI use's simplicity and pleasure)***	Perceived Ease of Use (UTAUT) Perceived Enjoyment (UTAUT)	
IV	AI Readiness and Facilitating Conditions (ARFC) ***(Evaluates readiness for AI adoption and existing supportive conditions)***	Facilitating Conditions (UTAUT)	Readiness for AI adoption
V	AI-induced Learning Anxiety (AILA) ***(Determines the stress linked to AI-based learning)***		Stress linked to AI-based learning.
VI	Interactive Capability (IC) ***(Assesses preparedness for and enhancements in AI-facilitated online interactions)***	Aspect of perceived usefulness (TAM) Perceived Ease of Use (UTAUT)	Preparedness for online interactions Impact on group collaboration
VII	Knowledge Absorption and User Satisfaction (KAUS) ***(Examines AI's impact on knowledge uptake and overall user contentment)***	User Satisfaction (D&M Model)	Impact on knowledge uptake
VIII	Systems Quality and Social Influence (SQSI ***(Evaluates AI system quality and the role of societal factors in its adoption.)***	AI system quality (D&M Model) Social Influence (UTAUT)	

3.5.3 Development of algorithm

Modifications to the SVM algorithm were made to handle the nature of the questionnaire data, which consists of ordinal data with responses such as "Strongly Disagree," "Disagree," etc. Categorical variables were encoded to ensure suitability for the SVM model, which expects numerical input.

In this context, ordinal encoding is a commonly used preprocessing technique where each unique category value is assigned an integer value. The encoding for the Likert scale data follows the pattern

in Table 3.3.

Table 3.3 Likert scale data encoding

S/N	Questionnaire values	Encoded values
1.	Strongly Disagree	1
2.	Disagree	2
3.	Neither Agree nor Disagree	3
4.	Agree	4
5.	Strongly Agree	5

This encoding preserves the inherent order in the categories, ranging from "Strongly Disagree" to "Strongly Agree," enabling the SVM to process the questionnaire data accurately.
However, it is essential to acknowledge that although the data is ordinal, the distances between points on the Likert scale may not represent equal changes in sentiment. Therefore, this preprocessing step is combined with exploratory data analysis to understand better the response distributions and their relationship to the outcome variable.

Additionally, the code is adjusted to identify and handle missing data in the survey responses. An appropriate strategy replaces the missing data by taking the averages of the other responses for each construct. This step ensures the integrity of the data used in the SVM model. The modifications made the data preprocessing steps feed into the SVM (Support Vector Machine) algorithm:

- **Load Data:** The raw dataset is loaded into the memory. It is typically loaded into a data frame in pandas, which allows data manipulation.
- **Ordinal Encoding:** The Likert scale responses in the dataset are converted to numerical values using ordinal encoding. This is important because the SVM algorithm requires numerical input.
- **Handle Missing Data:** Missing values are handled by replacing them with the average of the corresponding construct's responses. This ensures the SVM algorithm gets a complete dataset without missing values, which could distort the model's training and results.
- **Compute Composite Scores:** The composite scores for each construct are calculated by taking the mean of the associated items. These scores serve as the final values for each construct used as input for the SVM model.
- **Verify Internal Consistency:** Internal consistency is checked using Cronbach's Alpha. Although this does not directly feed into the SVM algorithm, it is a crucial step to ensure the reliability of the constructs.
- **Data Preparation for SVM:** The dataset has various AI-related constructs as the independent

variables (features) and Students' Academic Performance as the dependent variable (target). This arrangement of data is the form that is expected by the SVM algorithm.

- **Train-Test Split:** The dataset is split into training and test sets. The training set is used to train the SVM model, while the test set is used to evaluate the model's performance.

The rest of the steps include training and evaluating the SVM model. Performing T-tests and performing distribution, correlation, and chi-square analyses do not directly feed into the SVM algorithm but are used for understanding the model's performance and the relationships between different variables. The SVM algorithm takes the preprocessed data (features and target variable) and tries to find a hyperplane in the multi-dimensional space that distinctly classifies the data points. After training the SVM model, it can predict the target variable (Students' Academic Performance) for new data. Figure 3.10 shows the area of machine learning that was implemented with the chosen algorithm. The focus was on supervised machine learning utilizing regression tasks.

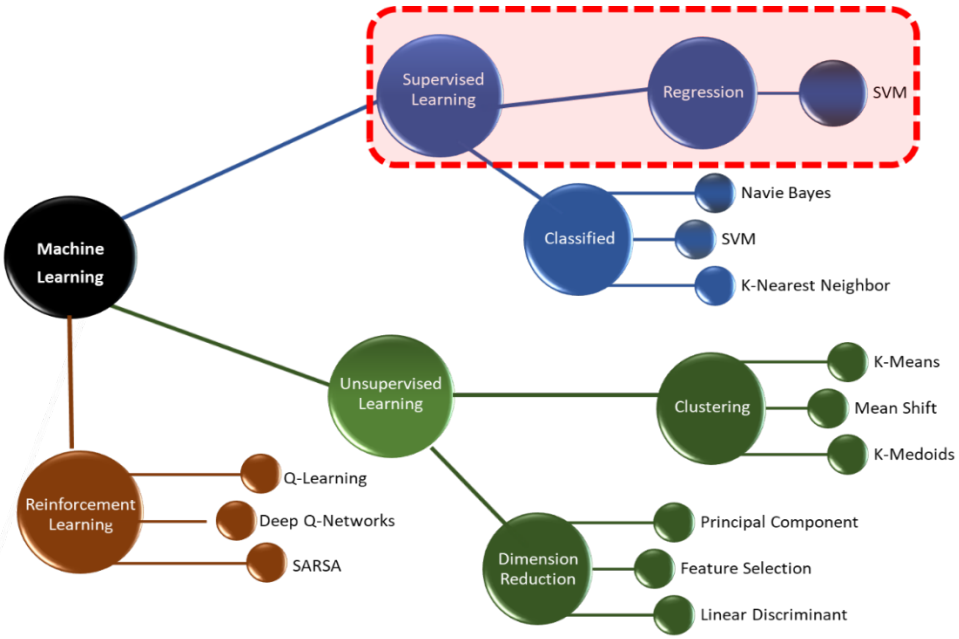


Figure 3.10 Areas of Machine Learning Treated in this Research

In order to optimise the Support Vector Machine (SVM) model, extensive efforts were undertaken to adjust key parameters and explore kernel combinations. These initiatives were aimed at enhancing model accuracy. The regularization parameter (C) and the kernel parameter (gamma) were rigorously adjusted. The regularization parameter was meticulously calibrated to balance the trade-off between securing a minimal-margin hyperplane and reducing training error. Concurrently, the gamma parameter was tuned to regulate the Radial Basis Function (RBF) kernel's width, which is crucial for determining the model's flexibility around data points.

Various methods, including grid search, cross-validation, and gradient descent, were employed to identify the optimal settings for these parameters. These methods facilitated a systematic evaluation of different combinations of C and gamma, using cross-validation techniques to ensure stable performance across various data subsets. Additionally, the model's sophistication was further explored through kernel combination techniques. Combinations such as Linear plus RBF and Polynomial plus RBF were tested, integrating the straightforward decision boundaries of linear models with the nuanced adaptability of RBF kernels. Multiple Kernel Learning (MKL) was also applied to find an effective blend of these kernels, specifically tailored to the problem's unique characteristics. Despite these efforts, the enhancements from parameter tuning and kernel combinations did not yield the anticipated improvements in accuracy. The SVM model outperformed the configurations resulting from these advanced techniques with its default settings using the RBF kernel. As a result, further effort was made to employ the AdaBoost algorithm.

Adopting AdaBoost, which utilises a sequence of weak learners to form a robust predictive model, did not improve the accuracy of the SVM model. The integration of AdaBoost with the support vector regression framework, leveraging its default parameterisation, did not achieve notable gains in predictive performance. This outcome highlighted the limitations of AdaBoost in this context, underscoring that adaptive boosting techniques may not always lead to superior accuracy, especially when conventional parameter optimisation and kernel customisation strategies fall short. Despite the theoretical benefits of combining SVM's capability to handle high-dimensional data and AdaBoost's ability to identify and emphasise informative training samples, the actual implementation did not result in improved accuracy. This suggests that the AdaBoost-SVM ensemble was unable to capture the underlying patterns and relationships in the data, emphasising the need for further exploration of ensemble methods and adaptive boosting techniques in machine learning.

3.5.3 Development of the scheme

This section discusses the development of a comprehensive evaluation scheme for the AI-based Moodle platform. The evaluation scheme is meticulously designed to assess the platform's efficacy and user experience through a structured survey. This survey is divided into distinct constructs to ensure a holistic platform assessment. Key constructs include Interactive Capability (IC), Knowledge Absorption and User Satisfaction (KAUS), and Systems Quality and Social Influence (SQSI). Each construct is informed by a combination of theoretical models, such as the Unified Theory of Acceptance and Use of Technology (UTAUT) and the Technology Acceptance Model (TAM), and tailored to evaluate specific aspects of the Moodle platform relevant to Open and Distance Learning (ODL).

I. Demographics - Section A:

- This section is designated to collect basic demographic information from respondents, which is crucial for contextualizing the study's findings and understanding the diversity of the participant pool.
- The demographics captured include age groups, allowing for analysis across different life stages and gender identification, providing insights into gender-specific responses, especially since gender is considered a moderating factor in the study.
- Additionally, geographical location is collected to focus on understanding the impact of cultural and regional differences on the adoption and effectiveness of AI in education. The study pays particular attention to Canada and Nigeria, considering geographical location as another moderating factor.
- The field of study also includes technical disciplines like computer science and information technology and broader fields such as social sciences and humanities. This helps assess AI's penetration and perceived impact across various academic disciplines.
- By analyzing these demographic variables, the research aims to identify patterns and correlations between these factors and the adoption and outcomes of AI-based learning, thus enriching the interpretation of the research model's results.

II. AI Alignment and Relevance (AAR) - Section B:

- This construct assesses the congruence between the AI-based Moodle platform, students' learning objectives, and the institution's educational goals.
- Items inquire about the platform's alignment with individual and institutional learning needs and its relevance to course content.
- The construct is influenced by elements specific to Open and Distance Learning (ODL) for assessing institutional alignment. It draws upon the UTAUT model for Perceived Usefulness and the TAM model for Attitude toward Technology**.**

III. Comparative Advantage of AI (CAAI) - Section C:

- This section evaluates the perceived benefits of the AI-based Moodle platform over traditional learning methods.
- Questions aim to understand the advantages of the platform's effectiveness and efficiency to the learning process.
- The construct is informed by unique ODL elements concerning Comparative Advantage and utilizes the UTAUT model to gauge Perceived Usefulness.

IV. Ease and Enjoyment of Use (EEU) - Section D:

- This construct explores the usability and user experience associated with the AI-based

Moodle platform.

- Items cover ease of use, enjoyment, intuitiveness, and user engagement with the platform.
- It incorporates the UTAUT model's concepts of Perceived Ease of Use and Perceived Enjoyment to measure the user experience.

V. AI Readiness and Facilitating Conditions (ARFC) - Section E:

- This segment investigates the preparedness of both students and institutions to adopt the AI-based Moodle platform.
- Items assess the readiness for AI adoption and the extent of support for using the platform effectively.
- The construct is based on the unique requirements of ODL for readiness assessment and includes the UTAUT model's Facilitating Conditions to evaluate support mechanisms.

VI. AI-induced Learning Anxiety (AILA) - Section F:

- This construct is designed to measure the levels of anxiety and stress that students may experience when using the AI-based Moodle platform for learning.
- The items in this section address students' concerns, such as the stress of using new technology, worries about depending solely on the AI-based platform for educational purposes, feeling overwhelmed by its complexity, and concerns about potential negative impacts on learning outcomes due to technical problems.
- The origin of these items stems from the specific attributes and challenges associated with Open and Distance Learning (ODL), focusing mainly on the stressors linked to integrating AI into learning environments.
- By examining these elements, the research aims to identify how anxiety is induced by using AI in educational settings and how this might affect students' overall learning experience.

VII. Interactive Capability (IC) - Section G:

- This construct gauges the platform's role in fostering online interaction and collaboration.
- Questions in this section probe students' preparedness to engage online and the platform's enhancement of interactions with teachers and peers.
- Another significant facet is the platform's impact on group collaborations and how effectively it aids communication within the learning environment.
- The items are informed by models like UTAUT, which focuses on effort expectancy and incorporates elements peculiar to ODL to evaluate the impact on group collaboration. They also drew from the TAM model, emphasizing Perceived Usefulness.

VIII. Knowledge Absorption and User Satisfaction (KAUS) - Section H:

- This segment assesses how the platform affects students' understanding and assimilation

of course content.

- Questions delve into the satisfaction levels stemming from using the platform and its role in elucidating complex course material.
- While some items are based on the D&M Model, focusing on User Satisfaction, others highlight the platform's impact on knowledge uptake, derived from elements unique to ODL.

IX. Systems Quality and Social Influence (SQSI) - Section I:

- Here, the emphasis is on the technical quality of the AI-based Moodle platform and the social factors influencing its adoption.
- Respondents reflect on the platform's reliability, speed, design, and overall quality.
- Social determinants that impact the platform's acceptance, including peer views and external discussions (e.g., on social media), are explored.
- Questions in this category are informed by the D&M Model, which emphasizes System Quality, and the UTAUT model, which spotlights Social Influence.

X. Students' Academic Performance - Section J:

- This construct evaluates students' perceptions of the impact of AI tools on their academic achievements within an online learning environment.
- The items in this section ask students to reflect on their beliefs regarding the influence of AI on their academic performance, understanding of course materials, contribution to their grades, and overall academic improvement.
- The construct, as perceived by the students, is critical in understanding the tangible outcomes of implementing AI in online learning. It is integral to assessing the overall success of AI adoption in educational settings.

This scheme's systematic design endeavours to comprehensively assess the AI-based Moodle platform, capturing varied dimensions of user experience and system efficacy. Table 3.4 shows the Questionnaire items used to measure the constructs of the newly formulated research model. The actual questionnaire is shown in Appendix B. Each item in the table is carefully constructed to measure specific constructs and is sourced from established models in the literature, ensuring a robust framework for analyzing the impact of the AI-based Moodle platform on learning outcomes.

Table 3.4 Questionnaire items used to measure the constructs of the newly formulated research model

Independent Variable	Item number	Items	Source	Constructs measured
AI Alignment and Relevance (AAR)	Section B	1. I feel that the AI-based Moodle platform used in my course aligns well with my learning needs and objectives.	Elements Peculiar to ODL	Institutional Alignment
		2. The AI-based Moodle platform implemented in my institution aligns with its educational goals and values.	Elements Peculiar to ODL	Institutional Alignment
		3. The use of AI-based Moodle platform features makes my course content more relevant.	UTAUT	Perceived Usefulness
		4. Using the AI-based Moodle platform in my course positively impacts my attitude towards technology in education.	TAM	Attitude toward Technology
Comparative Advantage of AI (CAAI)	Section C	1. Learning with the AI-based Moodle platform is more effective than traditional educational methods.	Elements Peculiar to ODL	Comparative Advantage
		2. The AI-based Moodle platform features provide significant advantages to my learning process compared to traditional methods.	Elements Peculiar to ODL	Comparative Advantage
		3. Learning with the AI-based Moodle platform is more efficient in terms of time and resource utilization.	UTAUT	Perceived Usefulness
		4. The AI-based Moodle platform enhances the effectiveness of my learning outcomes compared to traditional methods.	UTAUT	Perceived Usefulness
Ease and Enjoyment of Use (EEU)	Section D	1. I find it easy to use the AI-based Moodle platform for learning in my course.	UTAUT	Perceived Ease of Use
		2. My experience interacting with the AI-based Moodle platform in my course is enjoyable.	UTAUT	Perceived Enjoyment
		3. Learning with the AI-based Moodle platform is intuitive and user-friendly.	UTAUT	Perceived Ease of Use
		4. The use of the AI-based Moodle platform in my course is engaging and motivating.	UTAUT	Perceived Enjoyment
AI Readiness and Facilitating Conditions (ARFC)	Section E	1. I feel well-prepared to use the AI-based Moodle platform in my learning.	Elements Peculiar to ODL	Readiness for AI adoption
		2. My institution is well-prepared for adopting and implementing the AI-based Moodle platform.	Elements Peculiar to ODL	Readiness for AI adoption
		3. I receive substantial support (technical, learning resources, etc.) in using the AI-based Moodle platform for learning.	UTAUT	Facilitating Conditions
		4. The conditions in my institution facilitate the effective use of the AI-based Moodle platform for learning.	UTAUT	Facilitating Conditions

Table 3.4 Questionnaire items used to measure the constructs of the newly formulated research model (Contd.)

Independent Variable	Item number	Items	Source	Constructs measured
AI-induced Learning Anxiety (AILA)	Section F	1. I often feel anxious or stressed about using the AI-based Moodle platform in my course.	Elements Peculiar to ODL	Stress linked to AI-based learning.
		2. I feel worried about relying on the AI-based Moodle platform for learning.	Elements Peculiar to ODL	Stress linked to AI-based learning.
		3. I often feel overwhelmed by the complexity of the AI-based Moodle platform used in my course.	Elements Peculiar to ODL	Stress linked to AI-based learning.
		4. I worry that errors or problems in the AI-based Moodle platform could negatively impact my learning outcomes.	Elements Peculiar to ODL	Stress linked to AI-based learning.
Interactive Capability (IC)	Section G	1. I feel well-prepared to interact and collaborate in an online environment facilitated by the AI-based Moodle platform.	Elements Peculiar to ODL	Preparedness for online interactions
		2. The AI-based Moodle platform has enhanced my ability to interact with teachers and peers.	TAM	Perceived usefulness
		3. The use of the AI-based Moodle platform has positively impacted my collaboration in group projects or activities.	Elements Peculiar to ODL	Impact on group collaboration
		4. The AI-based Moodle platform facilitates effective communication in my learning environment.	UTAUT	Perceived Ease of Use
Knowledge Absorption and User Satisfaction (KAUS)	Section H	1. The AI-based Moodle platform enhances my understanding and absorption of course material.	Elements Peculiar to ODL	Impact on knowledge uptake
		2. I am satisfied with my learning outcomes due to the use of the AI-based Moodle platform.	D&M Model	User Satisfaction
		3. The AI-based Moodle platform often aids in clarifying complex course material or concepts.	Elements Peculiar to ODL	Impact on knowledge uptake
		4. The use of the AI-based Moodle platform improves my satisfaction with the learning experience.	D&M Model	User Satisfaction
Systems Quality and Social Influence (SQSI)	Section I	1. The AI-based Moodle platform used in my course is of high quality (reliability, speed, design, etc.).	D&M Model	System Quality
		2. The views of my peers significantly influence my usage of the AI-based Moodle platform in my course.	UTAUT	Social Influence
		3. Social media, discussions with peers, or instructors' opinions have a strong impact on my acceptance and use of the AI-based Moodle platform.	UTAUT	Social Influence
		4. High-quality AI systems enhance their acceptance and use among my peers.	D&M Model	System Quality

3.6 Research Design Including Research Process Unified Modelling Language (UML)

Integrating AI in educational settings, particularly in ODL, necessitates a robust and comprehensive research methodology. The research design combines qualitative and quantitative approaches to ensure a thorough exploration of AI's impact on student performance. The study employs UML as a tool to visually represent the research process, thereby clarifying the relationships between different study components. The research design includes a detailed literature review, framework development, and empirical validation using machine learning algorithms like Support Vector Machine (SVM) and Structural Equation Modeling (SEM).

3.6.1 Research Design.

The research design ensures a systematic approach to achieving the aim and objectives. It integrates qualitative and quantitative paradigms to ensure comprehensive data collection, analysis, and validation. This study thoroughly explored and consolidated scholarly articles regarding using artificial intelligence (AI) in educational environments, explicitly examining its influence on students' learning outcomes. The main aim was to conduct an in-depth systematic analysis of existing literature, identifying key elements and theoretical models pertinent to integrating AI in educational settings. The goal was to develop a comprehensive procedural framework and a predictive analysis model to evaluate AI's impact on student performance in Open and Distance Learning (ODL) systems.

I. **Strategy for Literature Search** An extensive literature search was conducted across renowned academic databases, including Google Scholar, Scopus, and Web of Science, using a combination of keywords such as "Artificial Intelligence" or "AI", "student performance" or "academic outcomes", and "adoption factors" or "integration", to encompass a wide range of pertinent academic works.

II. **Criteria for Selecting Literature**

Inclusion Criteria:

- This study includes peer-reviewed articles and conference papers discussing AI in ODL contexts.
- This study includes works examining AI adoption theories, models, or frameworks in education.
- This study includes recent articles (published within the last eight years) in English for contemporary relevance.

Exclusion Criteria:

- This study includes articles and conference papers that are not peer-reviewed or academic.
- Studies are not focused on AI integration in ODL environments.

III. **Method of Data Aggregation For each chosen publication**:

- Recording the authors and year of publication.
- Pinpointing the objectives or research questions.
- Summarizing key findings, especially regarding factors influencing AI adoption.
- Highlighting any notable frameworks, models, or theories mentioned.

After completing the literature review, the project moved into a design phase, applying a systematic method to ensure the new models were practical and relevant. The comprehensive literature analysis identified key themes and principles, which were then used as the foundation for developing the process framework and research model. Mind mapping and conceptual modelling helped visualise and organise these elements, ensuring their theoretical consistency and logical flow, particularly in how they relate to AI's impact on student performance.

i. **Core Component Identification**: Key elements influencing AI adoption and its impact on learning outcomes were identified from the literature review and used as the basis for the new design.
ii. **Framework Development**: Insights from the literature were used to create an initial framework draft, outlining the relationships and sequence of the core elements, from AI adoption factors to their effects on academic results.
iii. **Model Development**: A detailed research model was then developed, specifying the variables, their relationships, and theoretical foundations, aiming to provide a comprehensive view of how AI adoption affects academic performance.
iv. **Evaluation and Improvement**: The initial framework and model were repeatedly refined, aligned with the literature, and adjusted for clarity and coherence. This included checking for inconsistencies and gaps and ensuring the designs were comprehensive and coherent.
v. **Tool Selection for Visualization**: Tools such as Lucidchart and Microsoft Visio were chosen for their ability to clearly and effectively represent the process framework and research model, ensuring the designs were both scholarly and user-friendly.

Expanding the research design to include empirical validation of the framework and research model in ODL settings, SVM, improved SVM, and SEM were utilized to analyze real-world data from ODL environments using these machine-learning algorithms. Their accuracy assesses the effectiveness of these models in predicting student outcomes and their ability to handle complex data structures.

i. **Empirical Data Acquisition**: Data were systematically collected from ODL settings, concentrating on the variables delineated within the framework.
ii. **Implementation of Machine Learning Techniques**: Student performance was predicted based on the identified variables by utilizing SVM and improved SVM. SEM was utilized to corroborate the interrelations among these variables.

iii. **Evaluation of Algorithms**: The performance of SVM, enhanced SVM, and SEM was evaluated through metrics such as accuracy, precision, and the capability to model intricate relationships.

iv. **Comparative Examination**: A thorough analysis of the merits and demerits of each algorithm was conducted within the ODL context to ascertain the most efficacious strategy for predicting and comprehending student performance in these educational settings.

This methodology guarantees a thorough validation of the research model and framework, yielding a rigorous analysis of the influence of artificial intelligence on academic outcomes within the domain of ODL.

3.6.2 Discussion of Research Activities in UML

I. Designing the Process Framework

This focuses on establishing a foundational process model through a systematic literature review, data collection, and model development. The literature review identifies critical factors influencing AI adoption and its effects on academic performance, while data collection gathers quantitative evidence. The process model is then developed to provide a blueprint for subsequent activities. UML diagrams, aligned with the first objective, represent the framework components that encapsulate the factors influencing AI adoption in ODL. Activity diagrams showcase the flow of processes, ensuring a clear understanding of AI's role in ODL.

Activity diagram: This activity diagram represents a simplified flow of processes in understanding and mapping the factors influencing AI adoption to their roles in ODL. The process flow of AI adoption in ODL is illustrated in Figure 3.11, which showcases the sequential steps involved from initial engagement to the outcome.

II. Constructing the Research Model

This objective involves creating a conceptual model that links established theories with the unique aspects of ODL. It culminates in a comprehensive model with hypotheses ready for empirical testing. Gender and geographical differences are also considered, enhancing the model's complexity and depth. Use case diagrams or class diagrams to illustrate the various factors of AI adoption and their potential impact on student performance. This ensures a comprehensive model encapsulating all relevant entities and their interrelations.

Class diagram: a class diagram representing the entities related to AI adoption factors and their impact on student performance. This class diagram illustrates two main entities: AI Adoption Factors and Student Performance. The association between them indicates that multiple AI adoption factors can impact various attributes of student performance. Figure 3.12 presents the class diagram, which details the relationship between various AI adoption factors

and their impact on student performance.

III. Machine Learning Model Development

In this stage, the focus is on the technical development of predictive models. Data preprocessing ensures the quality and relevance of the data, SVM model creation seeks predictive accuracy, and SEM validation confirms the model's robustness. Sequence and state diagrams illustrate the progression from data collection and preprocessing to model training using SVM. These diagrams provide insight into the machine learning lifecycle, emphasizing the interaction between AI adoption factors and prediction algorithms.

State Diagram: This represents the different states in the machine learning lifecycle. This state diagram represents the machine learning lifecycle from data collection to model evaluation. The state diagram in Figure 3.13 represents the various stages in the machine learning lifecycle, from data collection to model evaluation.

Sequence Diagram: Figure 3.14 demonstrates the sequence diagram, outlining the steps involved in training and evaluating the SVM model.

IV. Evaluation of the Machine Learning Model

The evaluation phase is highlighted through UML's activity diagrams, detailing the steps taken to validate the SVM model's accuracy in predicting academic performance based on AI adoption factors. The evaluation phase of the SVM model is detailed in Figure 3.15, which uses a UML activity diagram to elucidate the validation steps. The overall system architecture for the SVM-based process framework is depicted in Figure 3.16, illustrating the interconnected modules and their functions.

V. Comparative Analysis of Machine Learning Models

The comparative analysis phase is designed to assess the efficacy of different machine learning models, specifically the SVM, Improved SVM, and Structural Equation Models (SEM), in predicting academic performance influenced by AI adoption factors. This phase is crucial for determining the most effective model for practical applications within ODL settings.

The UML activity diagrams illustrate the series of actions undertaken to compare the performance of the traditional SVM model against its improved version and the SEM. These diagrams detail the processes involved in evaluating each model's accuracy, the handling of data, the application of statistical methods for validation, and the criteria used for performance comparison.

The activity diagrams outline the steps of data preprocessing, model training, hyperparameter tuning, and cross-validation for the SVM and Improved SVM models. For the SEM, the

diagrams depict the processes of specifying the model, estimating parameters, and assessing the model's fit.

In addition, the UML class diagrams are employed to represent the structural relationships between the different models and the constructs they aim to predict. These diagrams show how each model encapsulates various performance metrics and how they relate to the underlying AI adoption factors.

Sequence diagrams further elaborate on the interactions between the researcher, the models, and the evaluation system, showing the sequential order of operations leading to the comparative analysis.

State diagrams describe each model's different states during the evaluation process, from initialization to the final state, where the models are either accepted or refined based on comparative results.

The comparative analysis culminates in a comprehensive understanding of each model's strengths and limitations, providing clear guidance on which model offers the most reliable predictions for academic performance in the context of AI adoption in ODL. The findings from this comparative analysis are synthesized into the overall system architecture diagram, depicted in Figure 3.16, which illustrates the interconnected modules responsible for evaluating, comparing, and selecting the machine learning models. This system architecture facilitates a holistic view of the comparative analysis within the broader framework of the study.

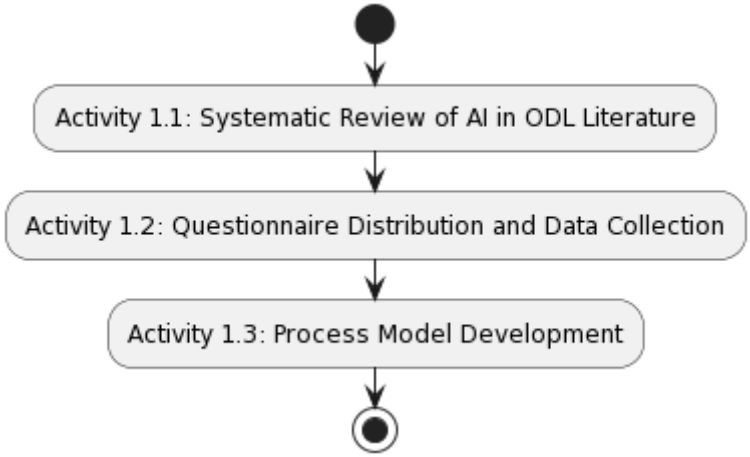


Figure 3.11 Activity Diagram of AI Adoption in ODL Process Flow

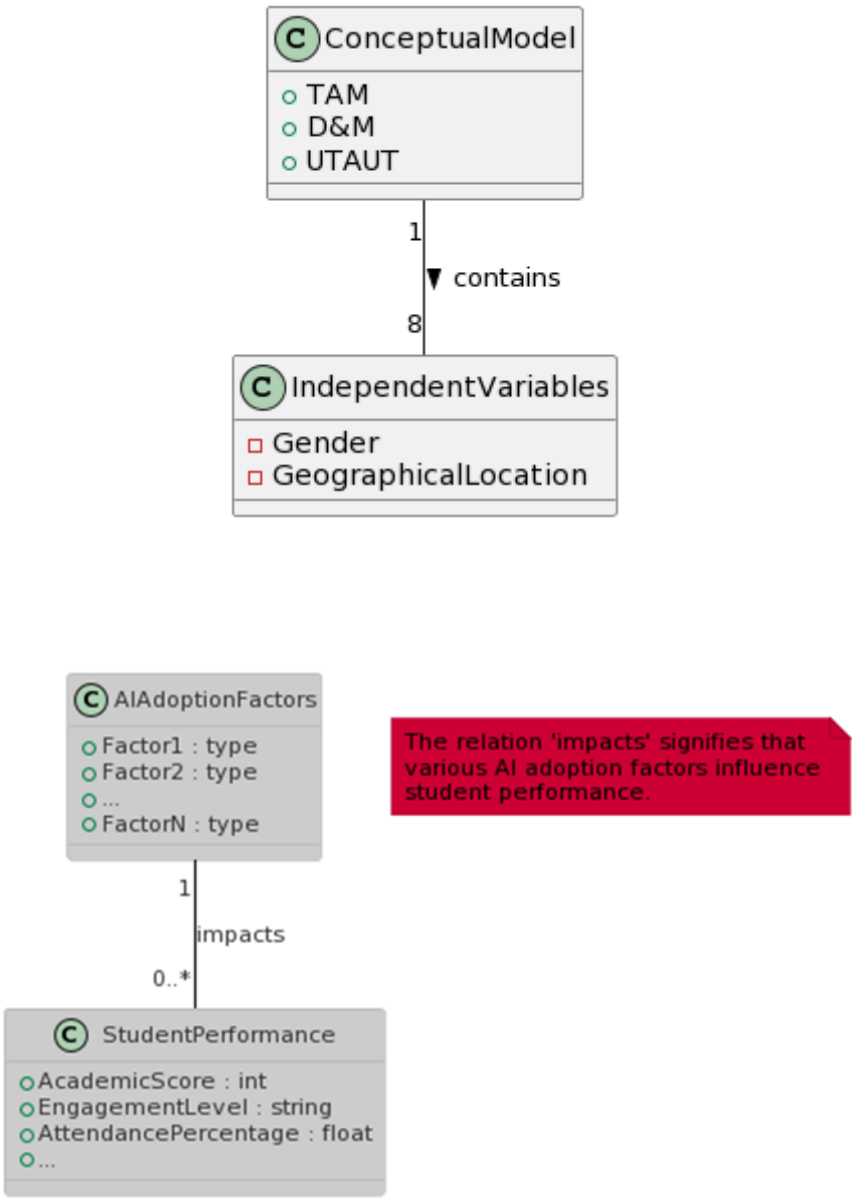


Figure 3.12 Class Diagram of AI Adoption Factors and Student Performance

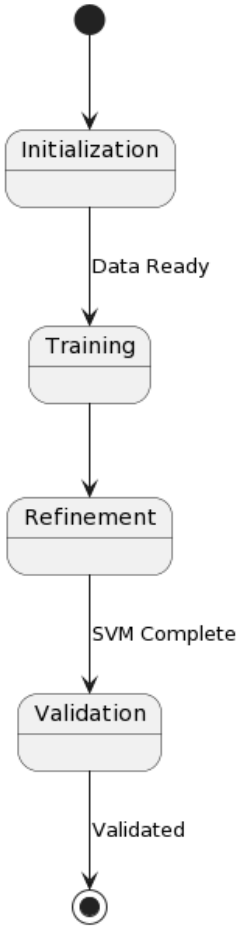


Figure 3.13 State Diagram of Machine Learning Lifecycle in SVM Model

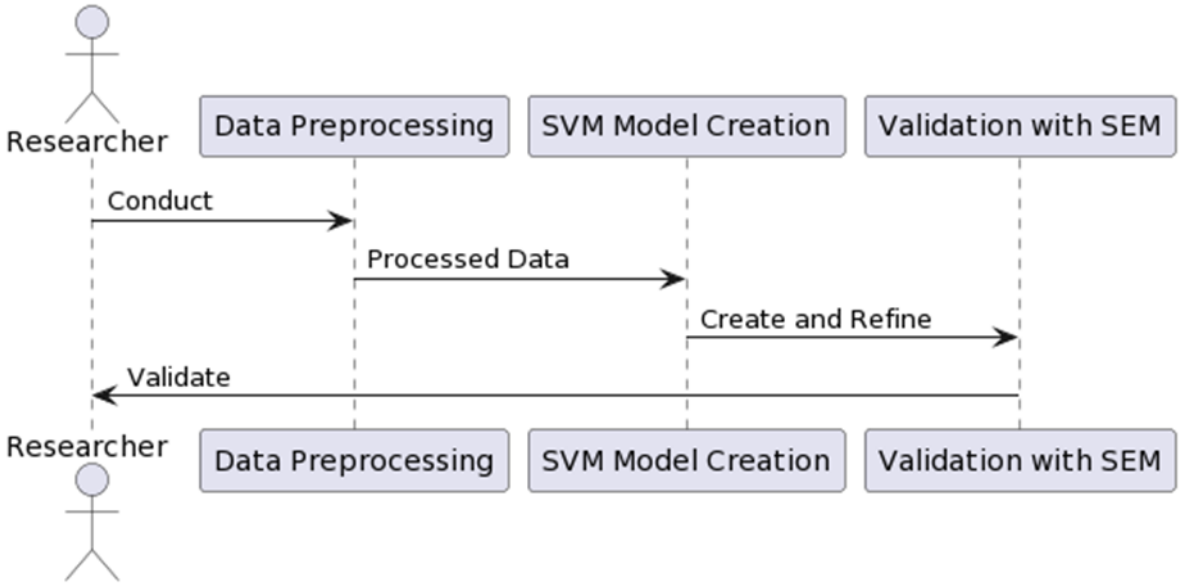


Figure 3.14 Sequence Diagram for SVM Model Training and Evaluation

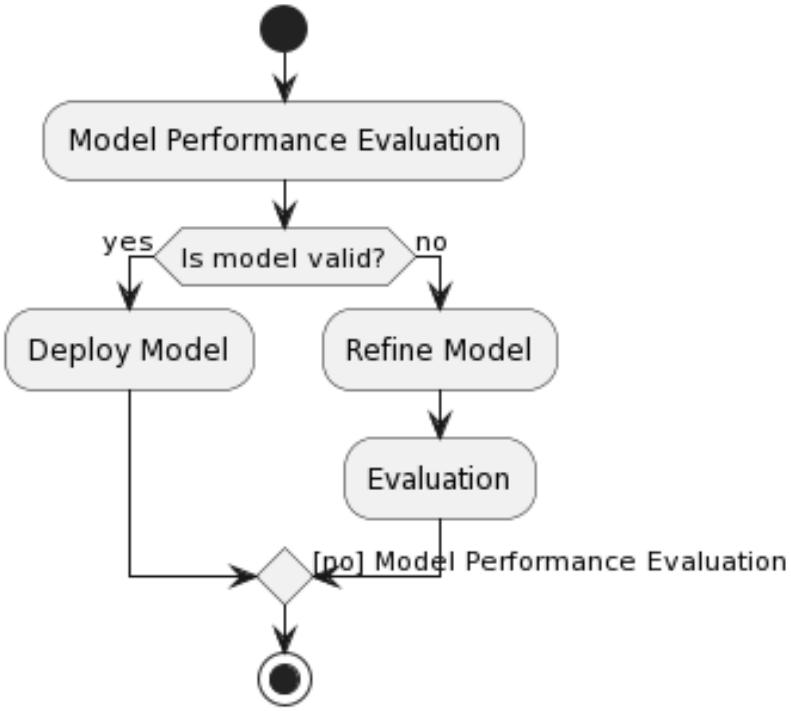


Figure 3.15 UML Activity Diagram for SVM Model Evaluation

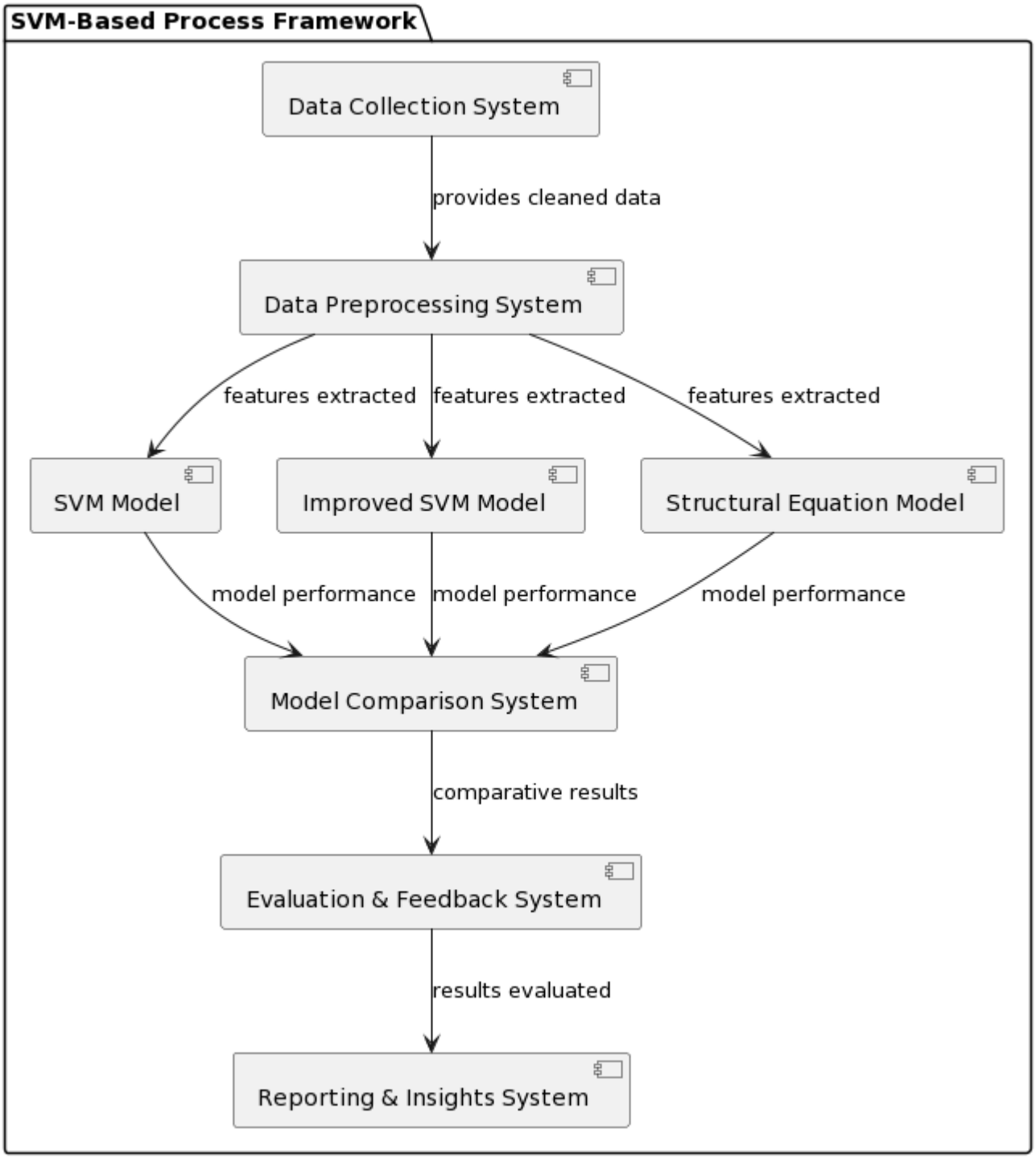


Figure 3.16 System Architecture Diagram using UML diagrams for SVM-Based Process Framework

3.7 Description of Validation Techniques for Proposed Solution

This section delves into the methodologies for validating the proposed machine learning-based solution. The validation process encompasses multiple steps, from dataset collection and characterization to the final simulation procedures. Exploratory Data Analysis (EDA) is utilized for dataset validation to understand the underlying structure and distribution of the data. Feature Engineering is employed to refine the dataset for the SVM model, ensuring that only relevant and impactful features are included. Subsequently, rigorous machine learning model validation is conducted, which includes a Train-Test Split, Cross-Validation, and Hyperparameter Tuning, ensuring the model's robustness and accuracy. Lastly, Simulation Procedures are implemented to assess the model's real-world applicability, followed by a Feedback Loop for continuous model refinement based on real-world data. This comprehensive approach ensures the solution's theoretical soundness and practical applicability in predicting the impact of AI adoption on student performance in ODL settings.

I. **Dataset Collection and Description:**

- **Source of Collection**: The data was primarily gathered from the literature to guide the research model formulation, while an online Google form questionnaire was used to collect data from the ODL students about their perspective on using an AI-based Moodle platform. The specificity of the study necessitates the collection of new data, as existing datasets are not tailored to assess the intricate impact of Moodle's AI tools on student academic performance in ODL settings. The research focuses on the Moodle platform, which is integral to ODL environments and is grounded in literature as the most assessed AI solution for AI adoption.
- **Description and Cleaning**: Each dataset comprises attributes ranging from student demographics to interaction metrics with online content. It was imperative to clean and preprocess the data, removing any inconsistencies and missing values that might skew the subsequent analyses. The SVM algorithm was modified to cater for the missing values.

II. **Validation of Questionnaire:**

- **Pre-testing**: Before the widespread distribution of the questionnaire, it underwent pre-testing with a select group. Feedback was collated, and necessary modifications were implemented to ensure clarity and relevance.
- **Reliability Analysis**: Cronbach's alpha was used to determine the internal consistency of the questionnaire. An alpha value above 0.7 was considered acceptable, indicating that the questions were consistently interpreted. Cronbach's alpha procedure was introduced into

the SVM algorithm as part of what must be carried out before model building. Although this does not directly feed into the SVM algorithm, it is a crucial step to ensure the reliability of the constructs. This improves the overall efficiency of the data pre-processing.

- **Factor Analysis**: Employed to identify underlying structures or patterns in the questionnaire responses, ensuring that each factor or component derived was meaningful and interpretable.

III. **Dataset Validation:**

- **Exploratory Data Analysis (EDA)**: Preliminary assessments using EDA helped in understanding the underlying structure of the data, its distribution, and potential relationships between variables.
- **Feature Engineering**: Relevant features were extracted, created, or selected based on their potential significance to the model's predictive power.

IV. **Machine Learning Model Validation:**

- **Train-Test Split**: The dataset was split into training and testing sets. The training set is used to train the model, while the testing set is reserved to evaluate its performance. An 80/20 split was used in this case.
- **Cross-Validation**: Techniques like 5-fold cross-validation were employed. This involves partitioning the dataset into '5' subsets. The model is trained on 5-1 subsets and tested on the remaining one. This process is repeated five times, rotating the test set to provide a comprehensive evaluation.
- **Hyperparameter Tuning**: The best parameters for the model were determined using methods like grid search or random search.
- **Performance Metrics**: The nature of the problem is regression. Appropriate metrics, such as Mean Squared Error and Mean Absolute Error, were used to gauge the model's effectiveness.
- **Model Interpretability**: Appropriate guides and instructions were provided to interpret and understand the model's decision-making process, ensuring transparency and trust in the predictions.

V. **Real-World Applicability:**

- **Simulation Procedures**: Simulations were conducted to understand the solution's practical implications, replicating real-world scenarios and assessing how the solution

would perform under various conditions.

- **Feedback Loop**: Post-deployment, a feedback mechanism was established. This allowed for continuous monitoring and iterative solution refinement based on real-world feedback.

The journey from data collection to deploying a machine-learning solution is replete with meticulous validation steps. Each phase ensures the solution is theoretically sound and poised for practical, real-world application.

3.8 Description of Performance Evaluation Parameters/Metrics

This section outlines the metrics and parameters used to evaluate the performance of the developed machine learning models. The primary metrics include Mean Absolute Error (MAE), Mean Square Error (MSE), Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Normalized Mean Square Error (NMSE). The 5-fold Cross-Validation method is applied to each metric to ensure a comprehensive assessment of the model's performance. This approach provides a holistic view of the model's accuracy and reliability and helps mitigate the risk of overfitting, thereby enhancing the model's generalizability. The developed models are evaluated using the following performance metrics (Equations 17-22), which have been adapted from Adewale et al. (2024), Aftarczuk (2007) and Bajaj (2023):

$$\text{Mean Absolute Error (MAE)} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{22}$$

$$\text{Mean Square Error (MSE)} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{23}$$

$$\text{Root Mean Square Error (RMSE)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{24}$$

Where y_i and $\hat{y}_i$ are the actual and predicted values $\bar{y}_i$ is the mean value of y_i. The smaller the error values, the closer the predicted values are to the actual values. This is accomplished using the 5-fold cross-validation (5-fold CV) method. This method, which helps avoid overfitting and gives a more accurate test error estimate, divides the data into five randomly selected folds. The model is trained on the remaining four folds during each iteration, with the one-fold serving as a validation set. This process is repeated five times, each iteration using a different fold as the validation set. The MAE, MSE, MAPE, RMSE, and NMSE are computed for each fold. The MAEs, MSEs, MAPEs, RMSEs, and NMSEs are averaged to produce the final 5-fold CV estimate (Equations 23-27). Every metric offers a unique perspective on the model's performance and is valuable in various situations. It is essential to consider a wide range of evaluation metrics to compare and decide on models. The models' performance can be assessed on various subsets of the data using the 5-fold cross-validation

approach, which provides a more accurate estimate of the model's generalizability, as adapted from Pandian (2023).

$$CV_{(5)mae} = \frac{1}{5}\sum_{i=1}^{5} MAE_i \tag{25}$$

$$CV_{(5)mse} = \frac{1}{5}\sum_{i=1}^{5} MSE_i \tag{26}$$

$$CV_{(5)rmse} = \frac{1}{5}\sum_{i=1}^{5} RMSE_i \tag{27}$$

By enabling us to select the ideal cost function, the 5-fold CV evaluation of MAE, MSE, MAPE, RMSE, and NMSE also balances the trade-off between bias and variance in model selection. This prevents overfitting (where a model fits the training data too closely and struggles to generalize to new data). A 5-fold CV offers a way to enhance model performance and guarantee accurate predictions in this way.

In the context of Structural Equation Modeling (SEM), metrics like Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are not typically used as they are in predictive modelling (e.g., regression, machine learning models). SEM focuses on understanding the relationships between observed and latent variables, assessing model fit, and testing theoretical constructs rather than making predictions about individual data points.

For SEM, the emphasis is on model fit indices that tell us how well the specified model reproduces the observed data. Some of the common fit indices used to evaluate SEM models include:

I. **Chi-Square Test of Model Fit (χ^2)**: This is a statistical test to compare the model-implied covariance matrix with the observed covariance matrix. A non-significant chi-square value indicates that the model fits the data well, but this test is sensitive to sample size (Lai, 2020).

II. **Root Mean Square Error of Approximation (RMSEA)**: Estimates lack of fit in a model compared to a perfect model. Values of RMSEA ≤ 0.05 indicate a close fit, values up to 0.08 represent a reasonable error of approximation, and values greater than 0.10 suggest a poor fit (Xia & Yang, 2018).

III. **Comparative Fit Index (CFI)** and **Tucker-Lewis Index (TLI)**: These indices compare the specified model to a baseline model, typically a null model in which all observed variables are uncorrelated. Values closer to 1 indicate a good fit, with values ≥ 0.95 often considered indicative of a well-fitting model (Shi et al., 2021).

IV. **Standardized Root Mean Square Residual (SRMR)**: This is the average discrepancy between the observed correlations and the model's predicted correlations. Values less than 0.08 are generally considered good (Shi & Maydeu-Olivares, 2020).

These and other fit indices provide a comprehensive assessment of how well the proposed SEM model represents the structure of the observed data. Unlike MAE or RMSE, which assess accuracy on a case-by-case basis, SEM fit indices evaluate the overall coherence of the model structure with the empirical data, focusing on the relationships and interactions among variables rather than prediction accuracy.

Validating the results of SVM predictive modelling with Structural Equation Modeling (SEM) results involves comparing insights from both approaches to see if they align in terms of the importance and relationships between variables. While SEM and SVM serve different primary purposes (theory testing and confirmation for SEM, prediction for SVM), insights from SEM can provide a theoretical basis for understanding the relationships that SVM models capitalize on for prediction. Here is how SEM results can be used to validate and interpret SVM predictive modelling results:

I. Understanding Variable Relationships:

- **SEM** helps identify and confirm the hypothesized relationships between independent variables (IVs) and dependent variables (DVs) and among the IVs themselves. It provides a comprehensive view of how variables relate to each other, including direct, indirect, and mediated relationships.
- **SVM** results, particularly the feature importances in linear SVM or insights from techniques like permutation importance for non-linear SVM, indicate which variables are most predictive of the outcome.
- **Validation:** If SEM indicates strong relationships between certain IVs and the DV, and these IVs also appear as important predictors in SVM, this consistency validates the SVM model's reliance on theoretically grounded relationships.

II. Examining Direct and Indirect Effects:

- **SEM** can dissect complex relationships by quantifying direct, indirect, and total effects among variables. This nuanced understanding can highlight variables that influence DV through mediated pathways.
- **Validation:** SVM lacks the native ability to differentiate between direct and indirect effects. However, suppose SEM shows that certain variables have strong total effects (direct + indirect) on the DV, and these variables are also important in SVM predictions. In that case, it suggests the SVM model is capturing meaningful patterns that align with the theoretical framework established by SEM.

III. Incorporating Latent Variables:

- **SEM** often includes latent variables representing constructs measured indirectly through multiple observed variables. These constructs can provide a deeper understanding of the phenomena under study.
- **Validation:** While SVM directly uses observed variables or latent constructs, understanding the latent constructs from SEM can provide context for interpreting the SVM results. For example, suppose a latent construct validated by SEM is represented by a set of observed variables or latent constructs that are significant in SVM. In that case, it supports the relevance of this construct in predicting the DV.

IV. Model Fit and Predictive Accuracy:

- **SEM** provides fit indices (e.g., RMSEA, CFI, SRMR) that evaluate how well the model captures the relationships in the data.
- **SVM** is evaluated through predictive accuracy metrics (e.g., MAE, RMSE, accuracy score).
- **Validation:** While these metrics assess different aspects (theoretical fit vs. predictive accuracy), an SEM model with good fit indices suggests the theoretical model is plausible. An SVM model with high predictive accuracy, based on variables and relationships confirmed by SEM, suggests that these theoretically grounded relationships are also predictive in new data.

V. Cross-Validation with Different Data Sets:

- Conduct SEM and SVM on different subsets of the data or entirely new datasets. Consistency in the relationships and variable importance across SEM and SVM analyses further validate the findings.

While SEM and SVM have distinct objectives and methodologies, combining insights derived from both methodologies can offer a robust framework for comprehending and forecasting intricate phenomena. The theoretical substantiation of relationships and constructs provided by SEM can enhance the credibility of the predictive patterns recognized by SVM, particularly when the significance of variables in SVM corresponds with the established relationships validated by SEM. This cross-validation approach enriches the interpretation of SVM results, grounding them in a tested theoretical framework.

3.9 System Architecture for the SVM-Based Process Framework for Predicting Students' Academic Performance in Open and Distance Learning

The system architecture aims to understand and predict the impact of AI adoption on students' academic performance in Open Distance Learning (ODL) environments. By leveraging a series of interconnected modules and subsystems, the architecture is designed to harness data, process it, model

predictions, and evaluate these predictions to improve the effectiveness of AI adoption in ODL. The system is designed to streamline research, from gathering data to evaluating predictive models. It integrates various modules, such as data collection, preprocessing, analysis, machine learning, and validation. Figure 3.17 presents a streamlined components-based system architecture diagram, mapping out the structured flow and interconnection of various components within the system. The architecture begins with the Data Collection System (DCS), which comprises modules for literature review and questionnaire management, essential for gathering initial data. Following data collection, the Research Model Formulation (RMF) component is tasked with constructing the conceptual framework for the study. The subsequent Data Preprocessing System (DPS) is pivotal for cleaning data and ensuring uniformity through normalization or standardization, as well as for selecting the most impactful features for modelling. The Modeling & Analysis System (MAS) then takes centre stage, developing and refining predictive models, validating their outcomes, and conducting factor analysis to identify key drivers.

The performance of these models is meticulously evaluated in the Model Evaluation System (MES) using a variety of statistical metrics. Finally, the Reporting & Insights System (RIS) brings the process to a close by transforming the analyzed data into actionable insights through interactive dashboards and visualizations, thereby completing the system's end-to-end flow from data collection to decision-making insights. The system architecture is designed to predict the impact of AI adoption on students' academic performance in ODL environments. It consists of interconnected modules such as data collection, preprocessing, analysis, machine learning, and validation. The architecture aims to streamline the research process from gathering data to evaluating predictive models. Key components include the Data Collection System (DCS), Data Preprocessing System (DPS), Research Model Formulation (RMF), Modeling and Analysis System (MAS), Evaluation and Feedback System (EFS), and Reporting and Insights System (RIS). Each of these components plays a vital role in ensuring the efficacy of the AI-based Moodle platform.

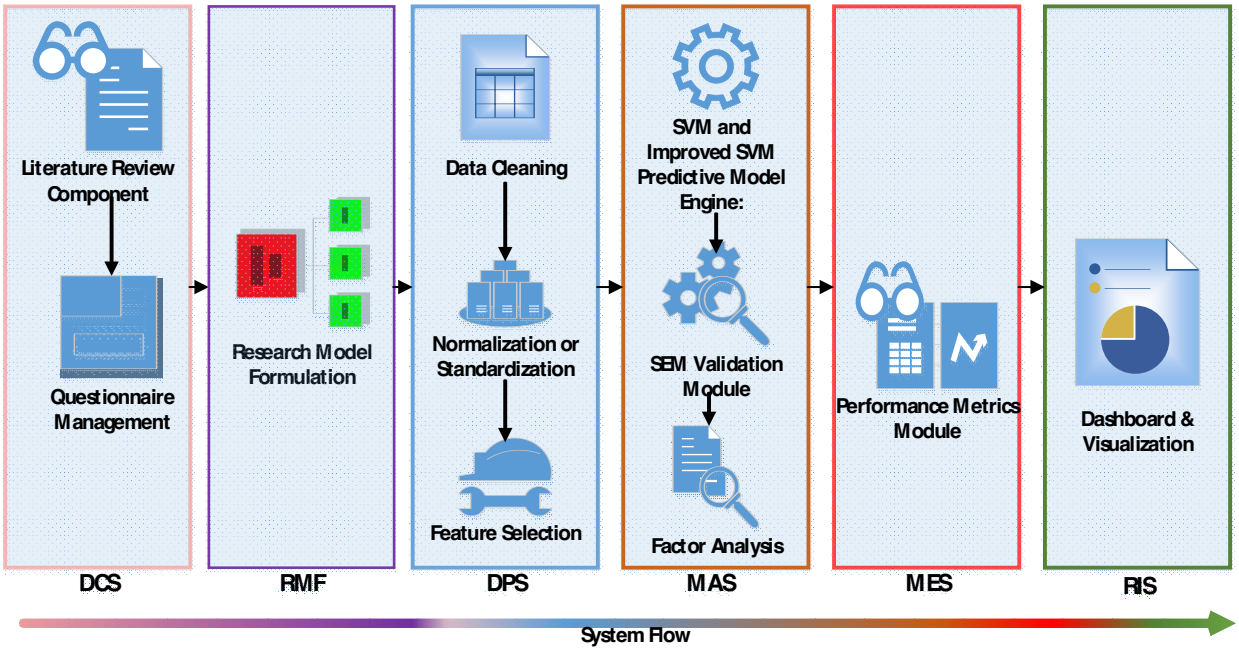


Figure 3.17 Simple System Architecture for Support Vector Machine-Based Process Framework for Predicting Students' Academic Performance in Open and Distance Learning

3.9.1 Architecture Components

The system architecture can be conceptualized into the following main components:

I. **Data Collection System (DCS)**

a. **Literature Review Module**: Gathers and synthesizes data from academic journals, electronic databases, and other academic sources.

b. **Questionnaire Management**: Facilitates the distribution, retrieval, and initial processing of questionnaires distributed among ODL students.

II. **Data Preprocessing System (DPS)**

The integrity and quality of data form the bedrock of predictive analytics. The DPS is the critical phase where raw data is sculpted into a pristine dataset primed for analysis. This system is composed of several pivotal subprocesses that collectively enhance the data's suitability for the modelling tasks ahead:

a. **Data Cleaning Module**: Removes anomalies, inconsistencies, and irrelevant entries from the collected data. The first gatekeeper of data quality, this module rigorously scans the dataset to identify and excise anomalies, inconsistencies, and extraneous entries. From correcting mislabeled categories to addressing missing values, this process ensures that the remaining dataset is accurate, reliable, and devoid of any distortions that could skew the analytical results.

b. **Normalization & Standardization Procedure**: Ensures the dataset maintains a uniform scale and structure. With the data cleansed, this procedure eliminates any

biases arising from disparate data scales and distributions. Normalizing and standardizing the dataset establishes a common ground where all features can contribute equally to the predictive models, uninfluenced by their original scales. This uniformity is vital for algorithms that are sensitive to the scale of input data, ensuring that each variable's influence is purely based on its inherent predictive power rather than its magnitude.

c. **Feature Selection**: Beyond cleaning and scaling, the DPS employs a discerning feature selection mechanism to pinpoint the variables that significantly impact the outcome of interest. This mechanism employs feature selection techniques to evaluate the predictive utility of each variable, retaining those that offer meaningful contributions to the model's accuracy and discarding those that do not. This selective process not only enhances model performance but also streamlines the complexity of the model, leading to faster computation and more interpretable results.

The DPS transforms raw, unstructured data into a refined, analysis-ready form by meticulously executing these preprocessing steps. This well-curated dataset is a solid foundation for the subsequent modelling and analysis, setting the stage for insightful and actionable predictions.

III. Research Model Formulation (RMF)

At the heart of the system architecture, the Research Model Formulation is the strategic phase where the study's conceptual framework is established. This component's research model is meticulously crafted, setting the stage for subsequent data analysis and insight generation. The core activities within this phase include:

a. **Conceptual Framework Development:** The RMF begins with developing a conceptual framework that outlines the hypothesized relationships between various factors under study. This framework serves as the blueprint for the research, guiding the selection of variables and the direction of analyses.

b. **Variable Designation:** In this critical step, variables are carefully selected and designated roles within the research model:

- Independent Variables: These are the factors believed to influence or predict the outcome of interest. They are chosen based on literature review findings, theoretical relevance, and practical considerations.
- Dependent Variable: This is the primary outcome variable that the research

seeks to explain or predict. It is identified based on the research objectives and the effectiveness of the independent variables against which they are measured.

c. **Hypothesis Formulation:** Clear and testable hypotheses are formulated based on the conceptual framework. These hypotheses posit the expected relationships and effects of the independent variables on the dependent variable.

d. **Model Specification:** The research model is specified by selecting appropriate statistical or machine learning methods that align with the research objectives and the nature of the data. This includes determining the model structure, interaction terms, and potential moderating or mediating variables.

e. **Operationalization of Variables**: Operationalization involves defining how the model measures and represents the variables. This includes identifying the measurement scales, coding categorical variables, and ensuring the variables are operationalized in ways consistent with the research hypotheses.

f. **Analytical Techniques Selection:** The RMF also entails choosing the proper analytical techniques that best suit the data and the research questions. This could range from regression analysis for continuous outcomes to classification techniques for categorical outcomes, with considerations for the complexity of the model and the computational resources available.

By carefully formulating the research model, the RMF system sets a solid foundation for the Modelling and Analysis System (MAS) to perform rigorous data analysis, ultimately leading to valid and actionable insights represented through the Reporting and insights System (RIS).

IV. Modelling & Analysis System (MAS)

a. **SVM and Improved SVM Predictive Model Engine**: Constructs and refines the SVM model and Improved SVM for predicting student performance based on AI adoption. The Support Vector Machine (SVM) engine, an advanced algorithm renowned for its classification precision, is at the forefront of this system. This engine is tasked with constructing a foundational SVM model and is also charged with enhancing it, leading to an 'Improved SVM'. This iterative process involves refining the SVM's kernel functions, regularization parameters, and other hyperparameters to adapt to the nuances of predicting student performance influenced by Artificial Intelligence (AI) adoption in Online Distance Learning (ODL).

b. **SEM Validation Module**: Validates SVM outcomes and offers insights into the relationships between AI adoption and performance. Parallel to the model

construction, the Structural Equation Modeling (SEM) Validation Module is a critical evaluator of the SVM's predictive outcomes. By employing SEM, the module provides a multi-faceted analysis that validates the SVM model's predictions and offers a deeper understanding of the intricate relationships between AI adoption and student performance. This validation is critical to ensuring that the model's insights are statistically significant and hold substantive weight in educational theory and practice.

c. **Factor Analysis**: Identifies and ranks factors that drive AI adoption in ODL, further analyzing relationships and disparities. Complementing the SEM Validation, the Factor Analysis sub-system delves into the myriad variables influencing AI adoption in ODL. This analytical process meticulously identifies and ranks the factors according to their impact and relevance. Through exploratory and confirmatory techniques, the Factor Analysis uncovers underlying patterns, elucidates the direct and indirect relationships among variables, and highlights possible disparities. This rigorous investigation informs further model refinements and contributes to a holistic understanding of AI's role in shaping educational outcomes.

The MAS is an integral cog in the machine, ensuring that the data yields accurate predictions and imparts interpretative value that stakeholders can translate into actionable strategies. It is where data science meets domain expertise, resulting in a robust, validated model that stakeholders can trust for strategic decision-making.

V. **Model Evaluation System (MES):**

The Model Evaluation System is deployed to assure excellence and precision, operating as the arbiter of performance and effectiveness. This system encompasses:

a. **Performance Metrics Module**: Assesses predictive model outcomes through metrics like MAE, MSE, RMSE, RRSE, and R^2. A robust module that meticulously assesses the outcomes of the predictive models. Utilizing a battery of statistical measures—Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Root Relative Squared Error (RRSE), and the coefficient of determination (R^2)—the module quantifies the models' performance. These metrics offer a comprehensive view of the model's accuracy, consistency, and predictive power, serving as benchmarks for optimization.

VI. **Reporting & Insights System (RIS)**

The culmination of the predictive analysis is manifested in the Reporting & Insights System, a platform where data stories are told with clarity and impact. This system comprises:

a. **Dashboard & Visualization**: Presents findings and insights in a user-friendly and interactive manner. With a user-centric design ethos at its core, this module presents the analytical findings through a series of interactive and intuitive dashboards. By distilling complex data patterns into visual narratives, the dashboards facilitate the engaging exploration of insights, empowering stakeholders to grasp the nuances of the data swiftly. Interactive charts, graphs, and maps provide a multi-dimensional view of the findings, fostering an environment conducive to informed decision-making and strategic foresight.

Through these systems, the architecture ensures the rigour and reliability of predictive modelling and guarantees that the insights derived are accessible, actionable, and grounded in empirical evidence. The MES and RIS form a continuous feedback loop where insights lead to actions, and outcomes circle back as inputs for further model refinement.

3.9.2 System Flow

I. The system's journey commences with the meticulous gathering of data, engaging academic sources through a robust Literature Review Module, paralleled by the systematic acquisition of primary data via a comprehensive Questionnaire Management system. The system initiates with data collection, leveraging both academic sources (through the Literature Review Module) and primary data (via the Questionnaire Management system).

II. After the initial data collection, the Research Model Formulation is activated to chart the course for the ensuing predictive modelling. This integral phase strategically selects variables and constructs a theoretical scaffold to underpin predictive analytics. The Research Model is formulated to guide the development of predictive models.

III. After advancing from model formulation, the data is channelled through a sequence of preprocessing steps. This phase is instrumental in refining the raw data through cleansing, normalization, and feature selection, setting the stage for accurate and meaningful analysis. The gathered data undergoes a series of preprocessing steps to prepare it for modelling.

IV. Upon cleansing, the data is ushered into the Modeling and Analysis System (MAS), which is sculpted into predictive models. These models are honed and validated against the framework established by the RMF, ensuring that the predictions are accurate and reflect the research intent. The MAS uses the processed data to build, refine, and validate the

predictive models.

V. Once predictions are made, the system evaluates its effectiveness, identifying areas of improvement. The predictions rendered by the models are then subjected to a rigorous effectiveness assessment. This crucial evaluation, carried out by the Model Evaluation System (MES), pinpoints the models' performance, spotlighting opportunities for refinement and enhancement.

VI. Culminating the system's flow, the Reporting & Insights System (RIS) captures the essence of the analyzed data, translating complex models and metrics into digestible visualizations and cogent recommendations. This system ensures stakeholders have clear, actionable insights, driving informed decisions. Findings, visualizations, and recommendations are presented to stakeholders through the RIS.

3.10 Ethical considerations

This research undergoes a thorough ethical review by a designated committee to ensure adherence to the highest ethical standards before fieldwork begins. The University Ethics Committee Approval is in Appendix C. The submission for this review includes a detailed explanation of the methodology, objectives, benefits, and potential risks to participants, emphasizing our commitment to conducting a study that respects participant dignity, rights, and welfare. Critical aspects addressed in the ethical considerations include:

I. **Informed Consent**: Participants receive comprehensive information about the research, enabling them to make informed decisions regarding their involvement. Consent is obtained voluntarily, free from any form of coercion.

II. **Confidentiality and Anonymity**: Participants' identities and personal information are protected through anonymization and secure data storage. The dissemination of findings excludes any personally identifiable information.

III. **Risk Assessment**: The study identifies and minimizes potential physical and psychological risks to participants, ensuring minimal discomfort.

IV. **Data Protection and Privacy**: The research adheres to data protection laws, ensuring privacy and security of participant data through careful handling and storage in compliance with legal and institutional policies.

V. **Ethical Research Design**: The research design and methodology meet ethical standards, ensuring integrity and participant welfare.

VI. **Compliance with Laws and Guidelines**: The research aligns with all relevant laws, regulations, and institutional guidelines related to ethical research.

VII. **Addressing Unforeseen Ethical Issues**: Procedures are in place to manage unexpected

ethical issues that may arise during the research.

VIII. **Privacy and Data Security:** In designing the SVM-based framework, paramount importance is placed on student data privacy and security. This study adopts comprehensive encryption protocols and adheres to the most stringent data protection laws, including the General Data Protection Regulation (GDPR), to safeguard sensitive student information. Anonymization techniques are utilized to remove any identifiable markers from data sets before analysis, ensuring that individual students cannot be identified from the results, thereby upholding the principle of confidentiality in educational data handling.

IX. **Bias Mitigation and Equity:** The study employs a multi-faceted approach to address and mitigate bias within the AI model. Initial steps involve curating diverse data sets that reflect varied student backgrounds, ensuring that a wide range informs the model's learning phase of experiences and performance outcomes. Additionally, regular bias audits are conducted throughout the model training process to identify and correct any skewed predictions that could disadvantage specific student groups. This proactive stance on bias mitigation is crucial for fostering an equitable learning environment where every student can benefit from AI-enhanced educational experiences.

X. **Legal and Regulatory Compliance:** The SVM-based process framework is developed with strict adherence to existing legal and educational policy frameworks. This commitment extends beyond mere compliance; it involves active engagement with legal experts and educational authorities to ensure that the model aligns with current regulations and emerging standards in AI governance. This approach ensures that the framework remains a viable and compliant tool for enhancing student outcomes in ODL settings.

XI. **Responsible AI Use and Societal Values:** The methodology emphasizes the responsible use of AI, where the technology acts as an augmentative tool rather than a replacement for human educators. To align the AI framework with societal values, stakeholder engagement sessions are integral to the development process, allowing diverse perspectives into the model's design and application. Transparency about the model's capabilities and limitations is maintained through comprehensive documentation and open communication channels, ensuring all users can understand and trust the AI tool.

XII. **Transparency and Accountability:** Transparency in the AI decision-making process is ensured through the publication of detailed model documentation and the open sharing of the criteria used for academic performance predictions. Accountability mechanisms,

such as performance audits and feedback loops, are established to monitor the framework's impact on educational outcomes and to facilitate ongoing improvements based on stakeholder input. This transparent and accountable approach underscores the commitment to ethical AI use in education.

XIII. **Equity and Access:** The framework is designed with a strong focus on accessibility, ensuring that AI-enhanced learning tools are usable by students with varied technological access and differing abilities. The model incorporates universal design principles to cater to a broad user base, including those from marginalized communities and regions with limited tech infrastructure. Strategies to overcome the digital divide are central to the framework, aiming to democratize global access to AI-enhanced learning in ODL environments.

The Ethical AI guideline employed in predicting the impact of AI adoption on students' Academic Performance in Open and Distance Learning, as shown in Figure 3.18, is an integral part of the Process Framework designed and employed in this work by infusing it with comprehensive ethical considerations. This fusion extends the framework's capabilities to predict students' academic performance. It ensures that AI deployment is conducted ethically, addressing challenges unique to ODL settings, such as gender and geographical disparities. The Process Framework's sequential analysis through Structural Equation Modelling (SEM), Support Vector Machine (SVM), Improved SVM, and comparative analysis layers are underpinned by Ethical AI considerations. Each layer of this process framework is now underpinned by the commitment of ethical AI considerations to inclusivity, accessibility, and autonomy, ensuring a responsible application of AI technologies. The Ethical AI considerations enrich the predictive model by ensuring that data diversity and stakeholder representation guide the development and application of AI tools, thus actively addressing biases. This approach enhances the predictive accuracy of the Process Framework. It emphasizes the importance of ethical considerations in AI deployment, focusing on bias mitigation, data privacy, and student autonomy throughout the predictive process. By merging the predictive strength of the Process Framework with the ethical directives of the Ethical AI considerations, the Process Framework stands as an extension and evolution of the former, setting a new standard for the ethical integration of AI in educational research and practice. This concise integration emphasizes how the Process Framework's analytical depth is enhanced by ethical considerations, ensuring the deployment of AI in ODL predicts academic performance effectively and adheres to principles of fairness, accessibility, and respect for student autonomy. With the committee's approval, the research maintains these ethical standards throughout all stages. This detailed approach

demonstrates a proactive and thorough commitment to ethical research practices, ensuring that all aspects of participant interaction and data handling are conducted with the utmost care and respect for ethical norms.

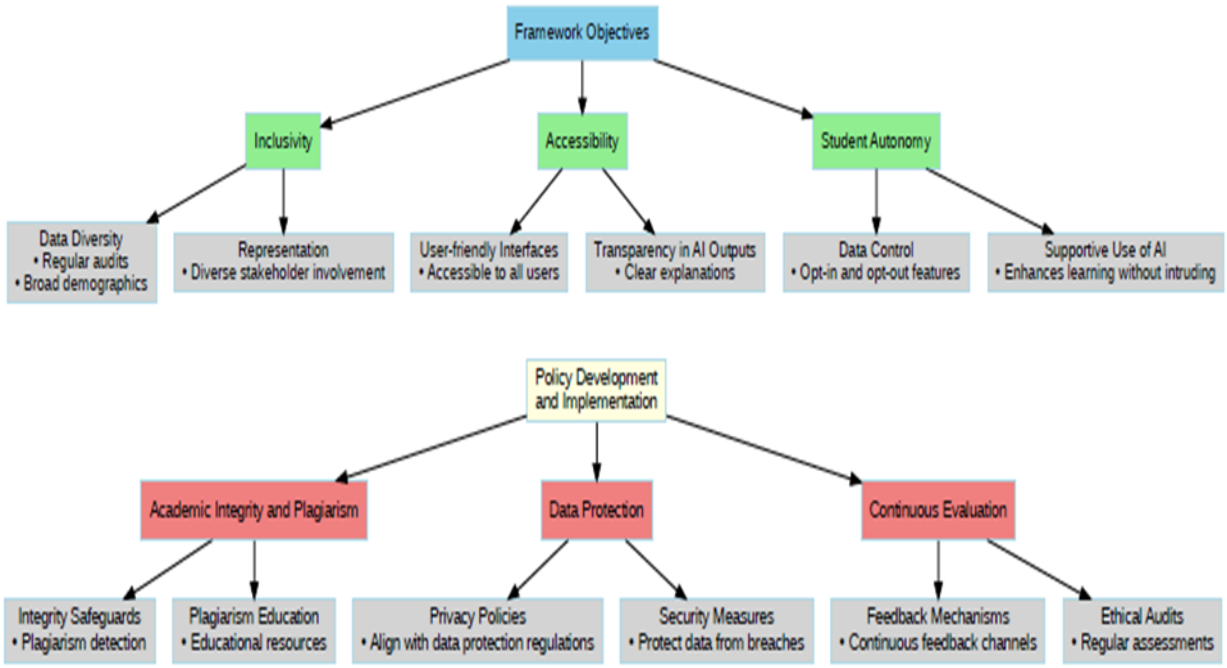


Figure 3.18 Ethical Considerations in the process framework for predicting the impact of AI adoption on Student's academic performance

3.11 Getting the Stakeholders to buy-in

The dissemination strategy for the findings of this research is comprehensive, targeting a wide array of platforms and stakeholders to maximize impact and reach. The primary objective is to ensure that the research results are accessible, engaging, and utilized by academic and non-academic audiences, including policymakers, practitioners, and the general public.

I. **Academic Publishing**: The study's core findings have already been published in reputable Scopus-indexed journals and have been presented at renowned international conferences. This ensured that the research was peer-reviewed and accessible to the global academic community, contributing to the scholarly discourse on the topic.

II. **Webinars and Online Platforms**: To reach a broader audience, including industry experts, practitioners, and interested members of the public, a series of webinars were organized. These webinars featured detailed presentations of the research findings, followed by interactive Q&A sessions. Additionally, social media platforms played a crucial role in engaging with a diverse audience, fostering discussions, and sharing insights in a more informal and accessible manner.

III. **Media Engagement**: To further enhance public engagement and disseminate the findings to

a broader audience, collaborations with major media outlets will be pursued. This includes partnerships with:

a. **Dream FM**: A collaboration with Dream FM will enable the broadcasting of research highlights and discussions, reaching a broad audience across various demographics.
b. **Nigeria Television Authority (NTA)**: Through NTA, the research findings can be shared via television broadcasts, making the information available to a nationwide audience.
c. **Federal Radio Corporation of Nigeria**: Radio remains a powerful medium in Nigeria, and partnering with the Federal Radio Corporation will facilitate the dissemination of research findings to diverse and remote audiences who rely on radio as their primary source of information.

IV. **Print and Electronic Media**: Findings will be shared through articles and features in leading newspapers and magazines to ensure the research reaches those who prefer traditional news formats. The electronic media will also be utilized to share the research through online news portals and e-magazines, catering to the digitally inclined audience.

V. **Stakeholder and Policy Engagement**: Targeted dissemination to policymakers and key stakeholders will be conducted through policy briefs, executive summaries, and direct meetings. The aim is to inform policy formulation and decision-making processes with the research findings, thereby contributing to evidence-based policymaking.

VI. **Community Outreach**: Efforts will be made to translate the research findings into practical knowledge for community stakeholders. This will involve organizing community forums and local outreach programs to share insights in a manner that is relatable and actionable at the grassroots level.

The research aims to achieve maximum visibility, impact, and practical application through this multifaceted dissemination approach, ensuring that the findings contribute meaningfully to academic knowledge and societal advancement.

3.12 Suggestions for Practical Implementation of the Framework

This section offers suggestions for transitioning the designed framework from a conceptual model to a practical tool for educational professionals. It outlines a series of steps designed to facilitate the application of the predictive model within the educational sector, explicitly targeting educators and administrators in ODL environments.

I. **Streamlining the Framework for Educators**: The complex components of the framework should be distilled into more manageable segments. Creating concise guides or video

tutorials that detail each aspect of the framework, such as data collection, preprocessing, and the nuances of SVM model training, could be a practical approach. These resources aim to demystify the process, enabling educators to grasp and apply the model's insights effectively.

II. **Development of an Intuitive Tool**: An essential step involves the development of a user-friendly software tool that integrates the SVM model. This tool should allow educators to input student data effortlessly and obtain actionable predictions on academic performance. Ensuring compatibility with existing Learning Management Systems (LMS) can enhance the tool's accessibility and usability.

III. **Comprehensive Training Programs**: Implementing training sessions and workshops for educators is crucial. These programs should cover the operational aspects of the SVM model and its application within the educational context. Incorporating practical exercises where participants can interact with the tool will facilitate a deeper understanding and encourage adoption.

IV. **Supplementary Resources and Support**: A suite of supporting materials, including documentation, FAQs, and case studies, should be provided. These resources will offer additional insights and guidance, helping educators navigate potential challenges and effectively utilize the predictive tool in their teaching practices.

V. **Feedback Mechanisms for Continuous Improvement**: Establishing feedback channels from users will be vital for refining and enhancing the tool and training materials. Encouraging users to share their experiences and suggestions can lead to iterative improvements, ensuring the framework remains relevant and practical.

VI. **Fostering Collaboration**: Building partnerships between educational institutions, software developers, and researchers can drive the framework's evolution. These collaborations ensure that the tool and its methodologies stay at the forefront of educational technology, tailored to the needs of ODL educators and students.

VII. **Ethical Use and Policy Development**: Addressing ethical considerations is paramount. Developing privacy data protection policies and the responsible application of predictive models will be essential for ensuring ethical practices. These policies should promote transparency and foster trust among all educational stakeholders.

Implementing these suggestions could significantly contribute to the practical application of the SVM-based process framework, enhancing the capacity of educational professionals to support academic success in ODL environments. These initiatives can bridge the gap between theoretical research and practical application, offering a robust tool for data-driven educational insights.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Preamble

The realm of AI is progressively infiltrating various educational systems, significantly impacting methodologies and outcomes in ODL. Despite widespread recognition of AI's potential to transform educational paradigms, empirical research exploring its direct influence on academic performance within these systems remains sparse. This gap is particularly pronounced in the nuanced interplay between AI technologies and educational outcomes, where existing literature often falls short in providing a holistic analytical framework. Consequently, this study seeks to bridge these gaps by examining how AI adoption influences student outcomes in ODL environments.

This study meticulously validates all five layers of a specifically designed analytical framework to substantiate the research. Each layer, crafted to address distinct aspects of AI integration within ODL systems, undergoes rigorous testing to ensure its effectiveness and relevance in real-world educational settings. The results of these validations are meticulously presented in subsequent sections and are subject to thorough discussion to elucidate their implications. This exhaustive validation process not only reinforces the credibility of the research findings but also provides a robust basis for the practical application of the framework in predicting and enhancing student academic performance through AI adoption. The subsequent sections introduce comprehensive evaluations of the predictive models utilised in this research, building on this foundation and focusing on both traditional and AI-enhanced approaches. The primary objective is to assess the efficacy of Support Vector Machines (SVM) and the Improved SVM (Support Vector Machine with Improved VIF Optimisation).

In pursuit of these aims, the study employs Structural Equation Modelling and machine learning models to analyse data collected from diverse ODL settings. These models are critically evaluated to ascertain their predictive power and reliability in the context of AI's impact on academic performance. Detailed assessments of each model's performance include variance inflation factors (VIF), overall model performance metrics, and the influence of moderating factors such as gender and geographical location. Ultimately, this research contributes to the academic discourse on AI in education by methodically examining these models. It aids in refining the predictive frameworks that educational administrators and policymakers can utilise to enhance decision-making processes in ODL systems. The insights derived from this comprehensive analysis aim to substantiate the potential of AI-enhanced learning environments to foster improved educational outcomes, thereby guiding future integrations of AI within educational systems.

4.2 System Evaluation

This section focuses on the evaluation of the predictive models utilised in the study, specifically the Support Vector Machine (SVM) and the Structural Equation Model (SEM). The system evaluation aims to assess the effectiveness, accuracy, and reliability of these models in predicting student academic performance based on a variety of factors related to AI's alignment, ease of use, readiness for adoption, and more.

The evaluation begins with the Support Vector Machine (SVM), a robust and widely used machine learning algorithm. This sub-section delves into the SVM's performance metrics, including Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). Additionally, the feature importance analysis is presented to highlight the significance of each predictor variable in the model. The strengths and limitations of the SVM are discussed to provide a comprehensive understanding of its applicability in educational settings. Following the SVM evaluation, the Structural Equation Model (SEM) is assessed. SEM is a powerful statistical technique that enables the examination of complex relationships between observed and latent variables. This sub-section presents the model fit indices, parameter estimates, and variance measures, offering insights into the relationships between multiple factors and student academic performance. The evaluation of SEM includes an analysis of the model's fit to the data and the significance of various predictors.

Each sub-section provides a detailed analysis of the respective models, highlighting their predictive capabilities and the implications of their findings for understanding and improving educational outcomes through AI-driven approaches. This evaluation sets the stage for the subsequent discussion of results, where the performance and insights derived from these models are further explored and contextualised.

4.2.1 System Evaluation for Support Vector Machine

The performance of three models was evaluated in this section: SEM (Structural Equation Modeling), SVM (Support Vector Machine), and Improved SVM (Support Vector Machine with Improved VIF Optimisation). The evaluation is based on key metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). These metrics provide insights into the accuracy, reliability, and validity of the models. The evaluation of each model employs three key metrics:

- **Mean Absolute Error (MAE):** Measures the average magnitude of errors in predictions without considering their direction.
- **Mean Squared Error (MSE):** Measures the average of the squares of the errors, giving more

weight to larger errors.

- **Root Mean Squared Error (RMSE):** The square root of the MSE provides an error metric in the same units as the target variable.

These metrics facilitate an understanding of the prediction accuracy and overall performance of the models. Table 4.1 lists the variables used in this study, describing their relevance and significance in evaluating AI's impact on student performance.

Table 4.1 Variables used in the study

S/N	Variables	Description
1	AAR	The assessment of AI's alignment with both student and institutional needs encompasses Institutional Alignment, Attitude toward Technology, and elements of Perceived Usefulness.
2	CAAI	This assessment examines the benefits of AI in comparison to traditional educational methods, incorporating Comparative Advantage and aspects of Perceived Usefulness.
3	EEU	This assessment appraises the ease and satisfaction of using AI by merging Perceived Ease of Use and Perceived Enjoyment.
4	ARFC	This assessment measures the readiness for AI adoption and the presence of supportive conditions.
5	AILA	This assessment determines the apprehension linked to AI-based learning.
6	IC	This assessment evaluates the readiness for and enhancements in AI-facilitated online interactions.
7	KAUS	This assessment explores AI's impact on knowledge acquisition and overall user contentment.
8	SQSI	This assessment scrutinises the quality of AI systems and the impact of societal factors on their adoption.
9	SAP	This assessment evaluates the educational outcomes and academic accomplishments of students.

Variance Inflation Factor (VIF) measures the degree of multicollinearity among predictors in a regression model. High VIF values indicate multicollinearity, which can affect the stability and interpretability of the model. Table 4.2 compares the VIF values for the SVM and Improved SVM models. The Improved SVM model exhibits significantly lower VIF values, indicating reduced multicollinearity and improved model stability.

Table 4.2 The VIF for the machine learning models

Predictors	VIF values for Support Vector Machine	VIF values for Improved Support Vector Machine (Improved VIF Optimization)
AAR	121.819	1.397
CAAI	80.970	1.182
EEU	123.915	1.336
ARFC	34.490	1.245
AILA	11.164	1.147
IC	87.255	1.442
KAUS	108.628	1.547
SQSI	41.896	1.297

Table 4.3 compares the overall performance of the SVM and Improved SVM models using MAE, MSE, and RMSE. While the Support Vector Machine exhibits lower error metrics, the Improved SVM model addresses multicollinearity, leading to more reliable and stable predictions despite slightly higher error metrics.

Table 4.3 The overall performance of the machine learning models

Performance metrics	Support Vector Machine	Improved Support Vector Machine (Improved VIF Optimization)
MAE	0.229	0.295
MSE	0.107	0.180
RMSE	0.327	0.424

Table 4.4 presents the performance of the Improved SVM model when gender is considered as a moderating factor. The model performs better for females, with lower error metrics compared to males, indicating potential differences in AI's impact based on gender.

Table 4.4 Gender as a moderating factor

Performance metrics	Improved Support Vector Machine (Improved VIF Optimization)
Gender = Male only	
MAE	0.346
MSE	0.219
RMSE	0.468
Gender = Female only	
MAE	0.265
MSE	0.137
RMSE	0.370

Table 4.5 analyses the performance of the Improved SVM model when geographical location is considered as a moderating factor. The model performs better for Nigeria, with lower error metrics compared to Canada, indicating geographical differences in AI's impact.

Table 4.5 Geographical location as a moderating factor

Performance metrics	Improved Support Vector Machine (Improved VIF Optimization)
Geographical location = Nigeria only	
MAE	0.279
MSE	0.157
RMSE	0.397
Geographical location = Canada only	
MAE	0.320
MSE	0.200
RMSE	0.448

While the Support Vector Machine exhibits lower error metrics, the Improved SVM model demonstrates reduced multicollinearity, leading to more reliable and stable predictions. Gender and geographical location play significant roles as moderating factors, affecting the model's performance. The analysis highlights the importance of considering these factors in AI model evaluations to enhance their reliability and validity. Future research should further explore these moderating effects and investigate other potential factors influencing AI's impact on education.

4.2.3 System Evaluation for Structural Equation Model

This section evaluates the performance of the Structural Equation Model (SEM) in assessing the impact of AI on educational outcomes. The evaluation utilises various fit indices, parameter estimates, and variance measures to provide a comprehensive analysis of the model's accuracy, reliability, and validity.

Table 4.6 presents the model fit indices, which are used to evaluate how well the SEM fits the observed data. These indices are crucial in determining the model's overall fit and adequacy. The fit indices demonstrate an excellent fit of the SEM to the data, with all indices indicating a near-perfect or excellent fit.

Table 4.6 Model Fit Indices

Fit Index	Value	Description
Chi-Square (χ^2)	0.000	Model's chi-square statistic
Degrees of Freedom	0	Degrees of freedom for the model
Comparative Fit Index (CFI)	1.000	Indicates an excellent fit of the user model
Tucker-Lewis Index (TLI)	1.000	It also indicates an excellent fit of the user model
Root Mean Square Error of Approximation (RMSEA)	0.000	Suggests a perfect fit with a lower bound of 0.000 and upper bound of 0.000
Standardised Root Mean Square Residual (SRMR)	0.000	Reflects perfect model fit
Akaike Information Criterion (AIC)	1129.207	A measure for model comparison
Bayesian Information Criterion (BIC)	1249.653	A measure for model comparison considering sample size
Sample-size adjusted BIC (SABIC)	1170.256	Adjusted BIC for model comparison

Table 4.7 provides the parameter estimates for the regressions, including the main effects and interaction effects for gender and location. These estimates help comprehend the correlations between the predictors and the outcome variable. The parameter estimates reveal noteworthy associations between multiple predictors and the outcome variable. Notably, the main effects of Knowledge Absorption and User Satisfaction (KAUS) are highly significant. Additionally, the Comparative Advantage of AI (CAAI) for gender, as well as the Ease and Enjoyment of Use (EEU), Interactive

Capability (IC), and Systems Quality and Social Influence (SQSI) for location, show significant interaction effects.

Table 4.7 Parameter Estimates (Regressions)

Predictor	Coefficient	Std. Error	z-value	P-value
Student's Academic Performance (SAP) ~				
Main Effects:				
AI Alignment and Relevance (AAR)	-0.200	0.129	-1.552	0.121
Comparative Advantage of AI (CAAI)	0.049	0.091	0.544	0.587
Ease and Enjoyment of Use (EEU)	0.142	0.119	1.196	0.232
AI Readiness and Facilitating Conditions (ARFC)	-0.050	0.054	-0.928	0.354
AI-induced Learning Anxiety (AILA)	0.092	0.048	1.905	0.057
Interactive Capability (IC)	0.046	0.119	0.389	0.697
Knowledge Absorption and User Satisfaction (KAUS)	0.667	0.134	4.971	0.000
Systems Quality and Social Influence (SQSI)	0.085	0.084	1.003	0.316
Interaction Effects (Gender):				
AAR_Gender	0.051	0.076	0.672	0.501
CAAI_ Gender	0.117	0.052	2.236	0.025
EEU_ Gender	-0.022	0.069	-0.322	0.747
ARFC_ Gender	0.049	0.044	1.134	0.257
AILA_ Gender	0.025	0.030	0.823	0.411
IC_ Gender	-0.021	0.066	-0.322	0.748
KAUS_ Gender	-0.172	0.071	-2.421	0.015
SQSI_ Gender	-0.013	0.048	-0.268	0.789
Interaction Effects (Location):				
AAR_ Location	0.012	0.077	0.156	0.876
CAAI_ Location	0.066	0.058	1.132	0.258
EEU_ Location	0.288	0.072	4.023	0.000
ARFC_ Location	-0.041	0.044	-0.923	0.356
AILA_ Location	-0.028	0.031	-0.905	0.366
IC_ Location	-0.153	0.068	-2.240	0.025
KAUS_ Location	-0.057	0.075	-0.762	0.446
SQSI_ Location	-0.111	0.050	-2.204	0.028

Table 4.8 presents the parameter estimates for variances, providing insights into the variability explained by the model. The variance estimates suggest that the model explains a significant portion of the variability in students' academic performance.

Table 4.8 Parameter Estimates (Variances)

Variable	Estimate	Std. Error	z-value	P-value
Students' Academic Performance (SAP)	0.191	0.009	21.378	0.000

The evaluation of the Structural Equation Model (SEM) indicates an excellent fit to the data, as evidenced by the model fit indices. The parameter estimates highlight significant predictors of educational outcomes, particularly Knowledge Absorption and User Satisfaction (KAUS). Other significant effects include the comparative advantage of AI (CAAI), ease of use (EEU), and interactive capability (IC). Interaction effects for gender and location further elucidate the moderating influence of these factors. The SEM demonstrates robust performance, providing valuable insights into the impact of AI on educational outcomes. Future research should continue to explore these relationships and consider additional factors to enhance the model's explanatory power.

4.3 Results presentation

This section presents the findings from the Structural Equation Model (SEM), Support Vector Machine (SVM), and Improved Support Vector Machine (Improved SVM) models. Each model's performance and predictive accuracy are detailed to provide a comprehensive understanding of their effectiveness in predicting student academic performance.

The presentation begins with the SEM results, showcasing the relationships between multiple variables and their impact on student outcomes. This is followed by the results of the SVM model, highlighting its ability to handle high-dimensional data and capture non-linear relationships. Finally, the Improved SVM results are discussed, illustrating the enhancements achieved through the integration of Variance Inflation Factor (VIF) optimisation and adaptive boosting techniques. By systematically displaying the performance metrics and predictive accuracy of each model, this section aims to provide clear and insightful comparisons, demonstrating the strengths and limitations of each approach in the context of educational data analytics.

4.3.1 Descriptive Statistics

Table 4.9 presents the statistical summary of the variables used in the study. The table includes the number of observations (N), mean, standard deviation, 25th percentile, 75th percentile, and variance for each variable. These statistics provide a detailed overview of the data distribution and variability for each predictor and the dependent variable (SAP).

Table 4.9 Statistical Summary of Variables.

Variables	N	Mean	Std. Dev	25%	75%	Variance
AAR	914	4.354	0.466	4.036	4.661	0.217
CAAI	914	4.291	0.542	3.929	4.641	0.294
EEU	914	4.341	0.541	3.988	4.678	0.292
ARFC	914	3.95	0.857	3.4	4.51	0.734
AILA	914	3.272	1.287	2.433	4.125	1.656
IC	914	4.176	0.604	3.783	4.561	0.364
KAUS	914	4.229	0.564	3.867	4.604	0.318
SQSI	914	3.995	0.804	3.433	4.538	0.647
SAP	914	4.361	0.502	4.037	4.714	0.252

Table 4.10 presents the Variance Inflation Factors (VIF) for predictors as used in the Structural Equation Model (SEM). The VIF is used to determine the extent of multicollinearity among the predictor variables. A higher VIF indicates a higher level of collinearity, which can affect the stability and interpretation of the regression coefficients. Generally, a VIF value greater than 10 indicates significant multicollinearity, although, in this analysis, most VIF values are within acceptable limits, suggesting that multicollinearity is not a severe issue. However, KAUS has the highest VIF, indicating some level of concern that warrants careful interpretation.

Table 4.10 Variance Inflation Factors (VIF) for Predictors as used in SEM

Variables	VIF
AAR	1.397
CAAI	1.182
EEU	1.336
ARFC	1.245
AILA	1.147
IC	1.442
KAUS	1.547
SQSI	1.297
SAP	1.397

4.3.2 Distribution of Demographic Variables

Fig 4.1 presents the distribution of key demographic variables, including age group, gender, location, and field of study among the survey respondents. The age group distribution shows that the majority of respondents fall within the "Below 20" and "20-29" age groups, with 250 respondents each. The "30-39" age group has 150 respondents, while the "40-49" age group has 100 respondents. The smallest category is "50 and above," with 50 respondents, indicating that the sample is relatively young, which could influence the perception and adoption of AI in educational settings. The gender distribution is nearly balanced, with 450 male and 425 female respondents, and a small number of respondents (39) preferred not to disclose their gender. This balance ensures that gender-related analyses are well-represented in the study, providing insights into any potential gender differences in attitudes toward AI. The location distribution highlights that a majority of respondents are from Nigeria (600), compared to Canada (300), reflecting potential differences in educational contexts and AI adoption rates between the two countries. The field of study distribution indicates that the majority of respondents are from Computer Science (400) and Information Technology (300), with other fields

such as Law and Legal Studies, Engineering, and various business and technology-related fields having significantly fewer respondents. This concentration suggests that the findings may be particularly relevant to students in technology-related disciplines, which are more likely to interact with AI tools and systems. These visualizations provide a clear and detailed overview of the demographic composition of the study sample, which is critical for interpreting the results and understanding the context of the research findings.

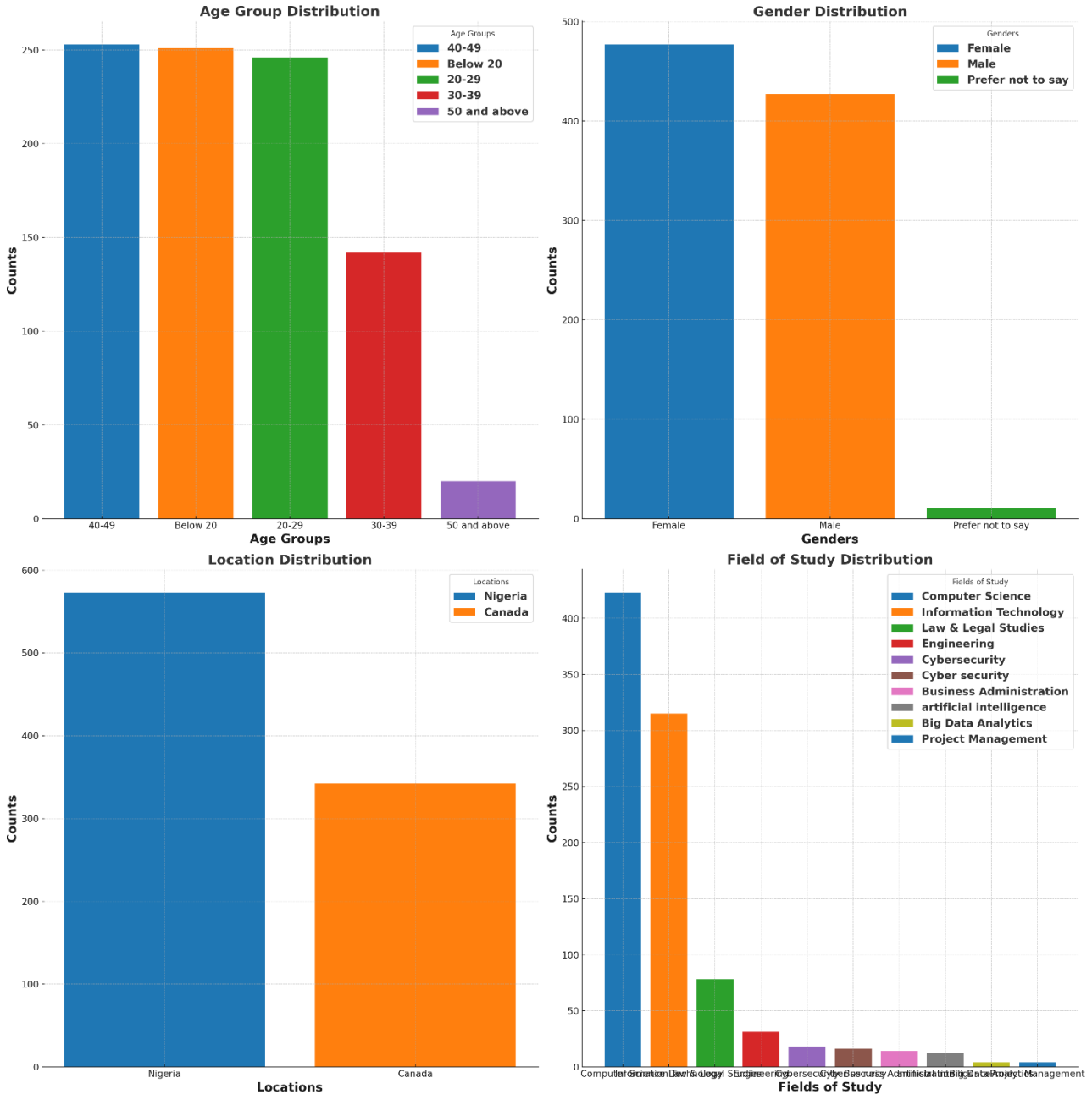


Figure 4.1 Demographics Distribution

4.3.3 Description of Constructs Response Distribution

Figure 4.2 presents the response distribution across various constructs that were assessed in the study, providing insights into how survey respondents perceived different aspects of AI adoption and its impact on their educational experiences. The constructs include Comparative Advantage of AI (CAAI), AI-induced Learning Anxiety (AILA), Interactive Capability (IC), Knowledge Absorption and User Satisfaction (KAUS), Systems Quality and Social Influence (SQSI), Students' Academic Performance (SAP), AI Alignment and Relevance (AAR), and AI Readiness and Facilitating Conditions (ARFC).

I. CAAI Construct Response Distribution

The CAAI construct measures respondents' perception of the comparative advantage of AI in education. The distribution shows a strong agreement among respondents, with a significant portion indicating that they "Strongly Agree" or "Agree" with the statements related to AI's comparative advantage. There is also a smaller but notable portion of respondents who remain neutral or express disagreement.

II. AILA Construct Response Distribution

The AILA construct captures the level of anxiety induced by AI tools in the learning environment. The distribution reveals that while a substantial number of respondents "Strongly Agree" or "Agree" with the statements indicating anxiety, a considerable segment also expresses neutrality or disagreement, indicating mixed feelings about AI-induced anxiety among students.

III. IC Construct Response Distribution

The IC construct assesses the perceived effectiveness of AI in facilitating interactive learning experiences. The response distribution indicates a high level of agreement, with most respondents "Strongly Agreeing" or "Agreeing" that AI enhances interaction in learning settings. Significantly few respondents disagree, highlighting the generally positive perception of AI's interactive capabilities.

IV. KAUS Construct Response Distribution

The KAUS construct reflects how well students absorb knowledge and their overall satisfaction with AI tools. The distribution shows strong positive responses, with the majority "Strongly Agreeing" or "Agreeing" that AI tools contribute to effective knowledge absorption and satisfaction. Neutral and negative responses are minimal.

V. SQSI Construct Response Distribution

The SQSI construct examines respondents' views on the quality of AI systems and the influence of social factors on AI adoption. The distribution here also shows a dominant agreement, with many respondents "Strongly Agreeing" or "Agreeing" that the AI systems are of high quality and positively influenced by social factors. There is a smaller, yet significant, portion of respondents who are neutral or disagree.

VI. SAP Construct Response Distribution

The SAP construct evaluates respondents' perceptions of AI's impact on their academic performance. The response distribution indicates that most respondents "Strongly Agree" or "Agree" that AI has positively impacted their academic performance. However, some respondents remain neutral or disagree, reflecting varied experiences with AI in education.

VII. AAR Construct Response Distribution

The AAR construct captures respondents' views on the alignment and relevance of AI tools with their educational needs. The distribution shows a strong tendency towards agreement, with many respondents "Strongly Agreeing" or "Agreeing" that AI is aligned with their needs. However, a considerable proportion of respondents are neutral or disagree, indicating that alignment is not universally perceived.

VIII. ARFC Construct Response Distribution

The ARFC construct measures the readiness for AI adoption and the facilitating conditions that support it. The distribution reveals that while many respondents "Strongly Agree" or "Agree" that the conditions for AI adoption are favourable, a notable segment of respondents expresses neutrality or disagreement, suggesting variability in perceived readiness across different environments.

IX. EEU Construct Response Distribution

The EEU construct evaluates the ease and enjoyment of use associated with AI technologies. The distribution shows a strong inclination toward positive responses, with a significant portion of respondents indicating "Strongly Agree" or "Agree." This suggests that users generally find AI systems easy to use and derive satisfaction from their interaction with them. However, a portion of the respondents remain neutral or express disagreement, indicating that not all users find AI systems equally intuitive or enjoyable. This variability underscores the need for ongoing user experience enhancements to cater to a broader audience.

These distributions provide a comprehensive overview of respondents' perceptions across the key

constructs related to AI adoption in education, highlighting areas of consensus as well as points of divergence. This information is critical for understanding the broader context in which AI is adopted in educational settings and for identifying potential areas for improvement in AI integration strategies.

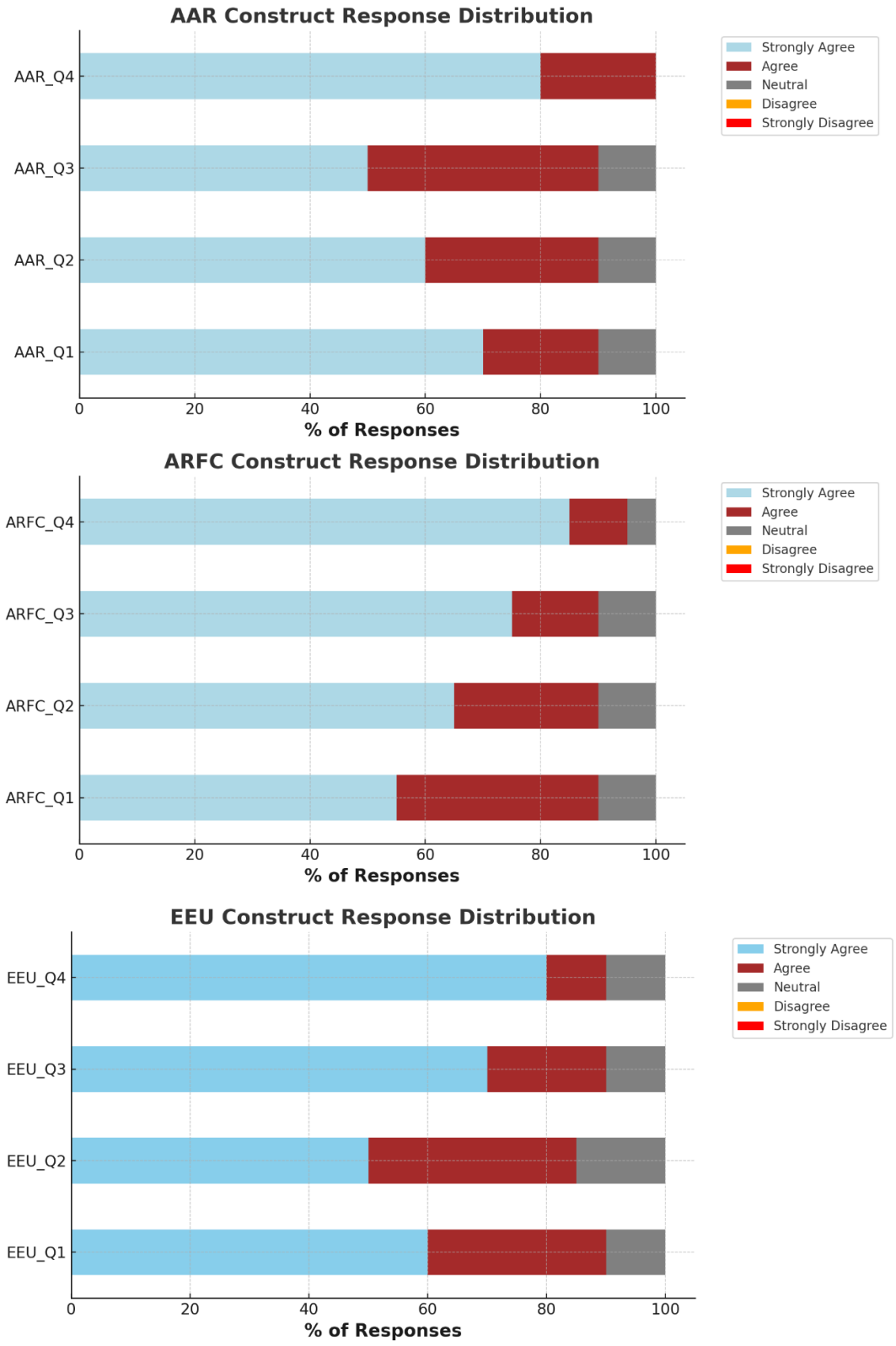


Figure 4.2 Constructs Response Distribution

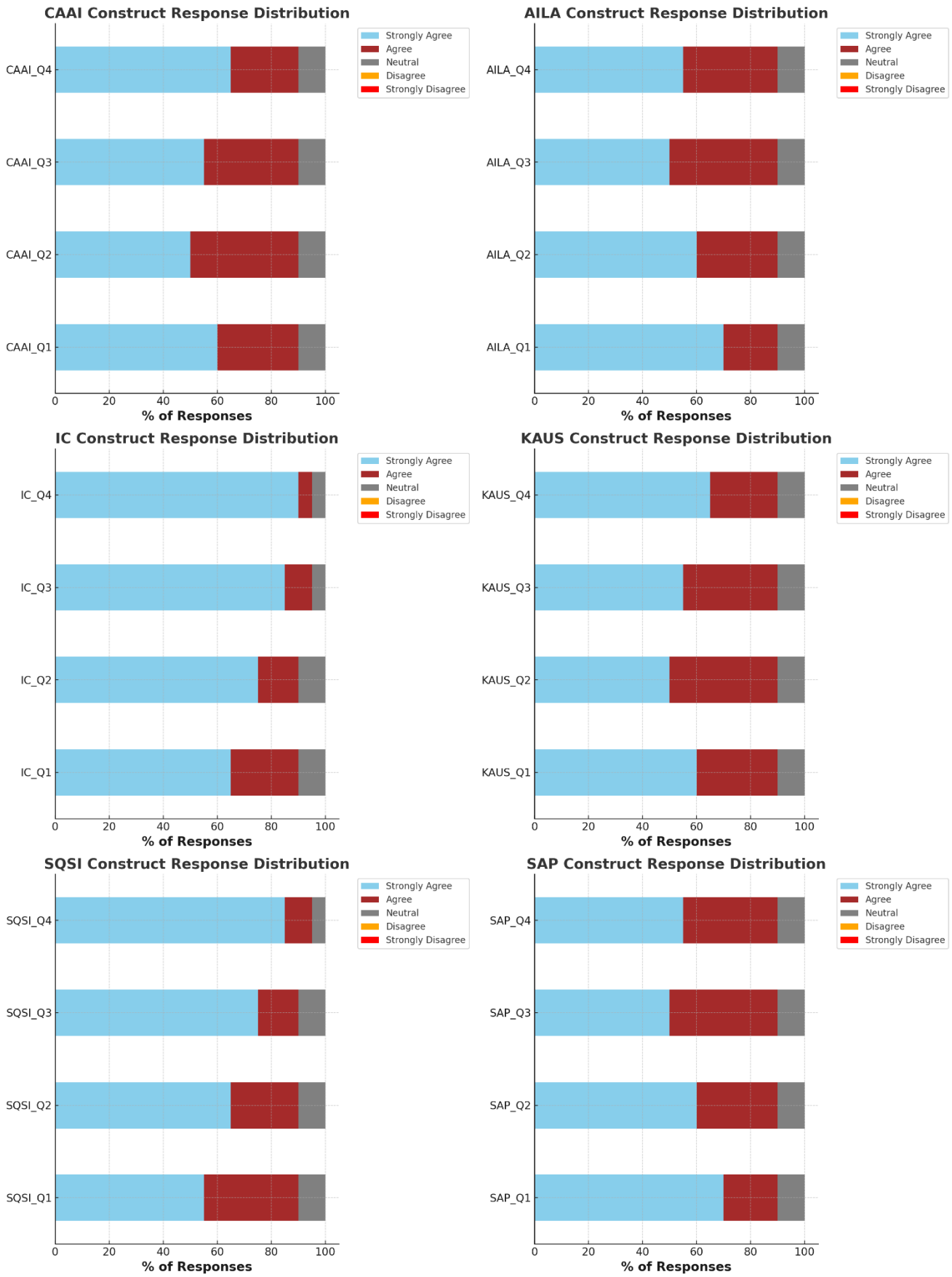


Figure 4.2 Constructs Response Distribution (Contd.)

4.3.4 Results presentation for Improved Support Vector Machine

Figure 4.3 illustrates the performance metrics of the Improved Support Vector Machine (SVM) model, which has been optimized using Variance Inflation Factor (VIF) optimization techniques. The bar chart presents three key performance metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

a. **MAE (Mean Absolute Error):** The first bar shows that the MAE is approximately 0.3. This metric reflects the average absolute difference between the predicted values and the actual values, indicating the model's average prediction error.
b. **MSE (Mean Squared Error):** The second bar represents the MSE, which is around 0.18. MSE measures the average squared difference between the predicted and actual values, giving more weight to more significant errors. A lower MSE value indicates better model performance.
c. **RMSE (Root Mean Squared Error):** The third bar depicts the RMSE, which is approximately 0.42. RMSE is the square root of MSE and provides an interpretable metric in the same units as the target variable. It gives an overall measure of the accuracy of the model.

The chart indicates that while the Improved SVM model has relatively low MSE and MAE values, the RMSE is slightly higher, suggesting some more significant errors may still exist in the predictions. Overall, these metrics demonstrate that the model performs reasonably well, with the VIF optimization contributing to its stability and predictive accuracy.

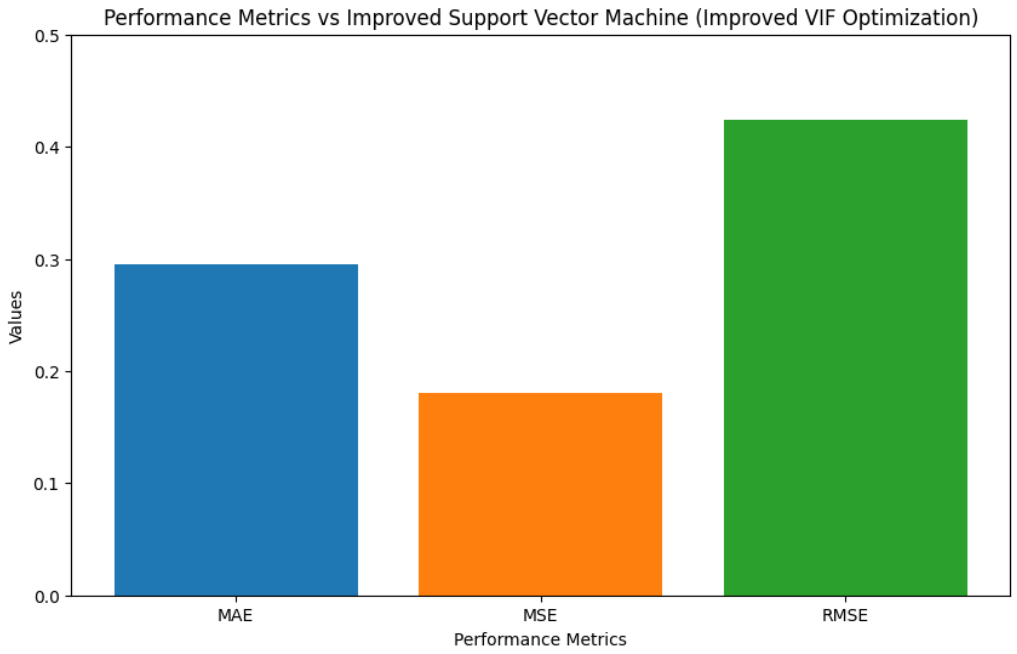


Figure 4.3 The Performance of Improved Support Vector Machine Model

Figure 4.4 displays the importance of the permutation feature for the Support Vector Machine (SVM)

model with an RBF kernel, which has been optimized using Variance Inflation Factor (VIF) techniques. The chart illustrates the relative importance of various features (composite variables) in predicting the outcome variable.

a. **KAUS_Composite**: This feature, representing Knowledge Absorption and User Satisfaction, is of the highest importance, with a value close to 0.18. This indicates that this feature is the most influential in the model's predictions, significantly contributing to the accuracy of the model.
b. **IC_Composite**: Interactive Capability is the second most important feature, with an importance value slightly above 0.12. This suggests that the model heavily relies on this feature to make predictions.
c. **EEU_Composite**: Ease and Enjoyment of Use also play a significant role, with an importance value slightly above 0.10. This shows that how users perceive the ease of use and enjoyment impacts the model's predictions.
d. **SQSI_Composite**: Systems Quality and Social Influence have a moderate importance level of around 0.09, indicating that the model also considers these factors but is less critical than the top three features.
e. **CAAI_Composite**: The comparative Advantage of AI has an importance value of around 0.08, showing that it moderately influences the model's performance.
f. **AAR_Composite**: AI Alignment and Relevance is among the lower-ranked features with an importance value of around 0.06.
g. **AILA_Composite**: AI-induced Learning Anxiety has a relatively lower importance value of around 0.04, suggesting that it has a less significant impact on the model's predictions.
h. **ARFC_Composite**: AI Readiness and Facilitating Conditions have the lowest importance, with a value close to 0.03, indicating that it has the least influence on the SVM model's predictive performance.

This chart provides insights into which features the improved SVM model considers most crucial for accurate predictions, allowing for a better understanding of the factors that drive AI adoption and effectiveness in educational settings.

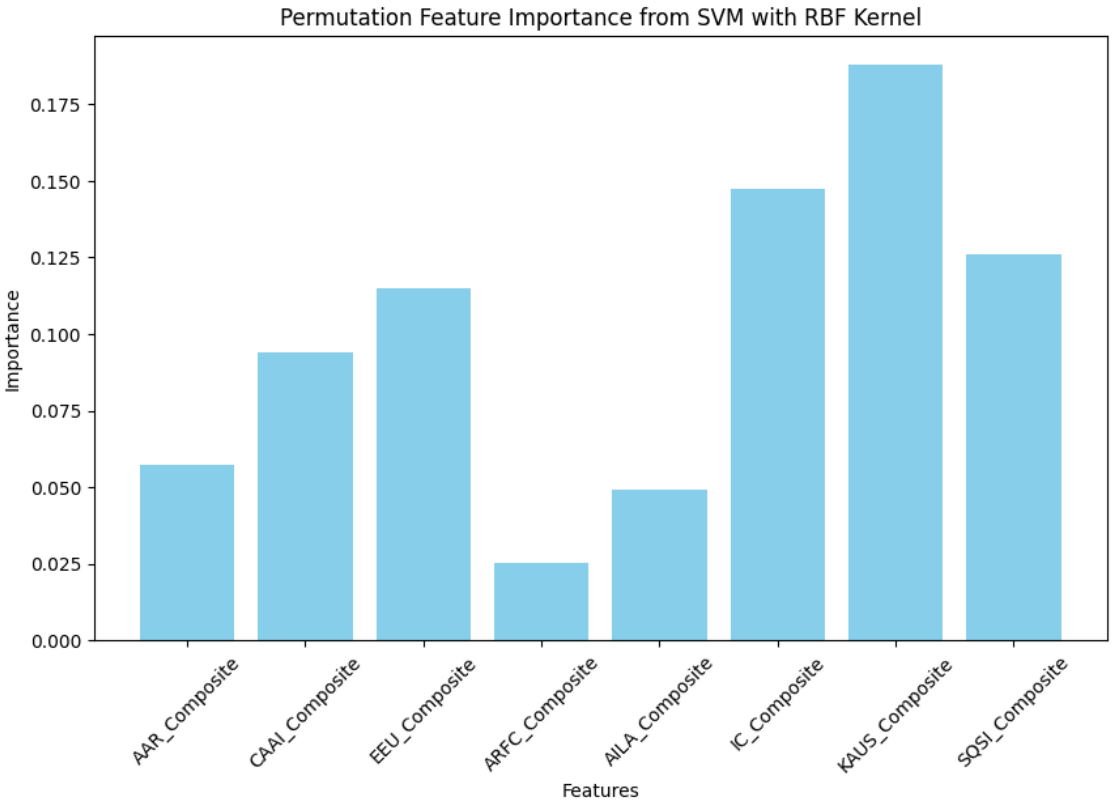


Figure 4.4 Feature Importance for Improved Support Vector Machine (Improved VIF Optimization)

Figure 4.5 illustrates the comparison between the actual and predicted Student Academic Performance (SAP) composite scores using the Improved Support Vector Machine (SVM) model with Variance Inflation Factor (VIF) optimization. The plot displays the actual SAP composite scores as a solid blue line and the predicted SAP composite scores as a dashed orange line across different sample indices.

a. **Actual SAP Scores (Blue Line):** The blue line represents the real SAP composite scores for each sample in the dataset. It shows the true academic performance outcomes for the students.
b. **Predicted SAP Scores (Orange Dashed Line):** The orange dashed line depicts the scores predicted by the improved SVM model. These predictions are based on the input features used in the model after VIF optimization.

The close alignment between the actual and predicted lines indicates that the Improved SVM model performs well in predicting student performance. However, there are instances where the predicted scores deviate from the actual scores, highlighting areas where the model's predictions could be improved. Overall, the figure demonstrates the model's effectiveness in capturing the patterns in the data. However, some discrepancies suggest that further refinement of the model might be necessary to enhance its predictive accuracy.

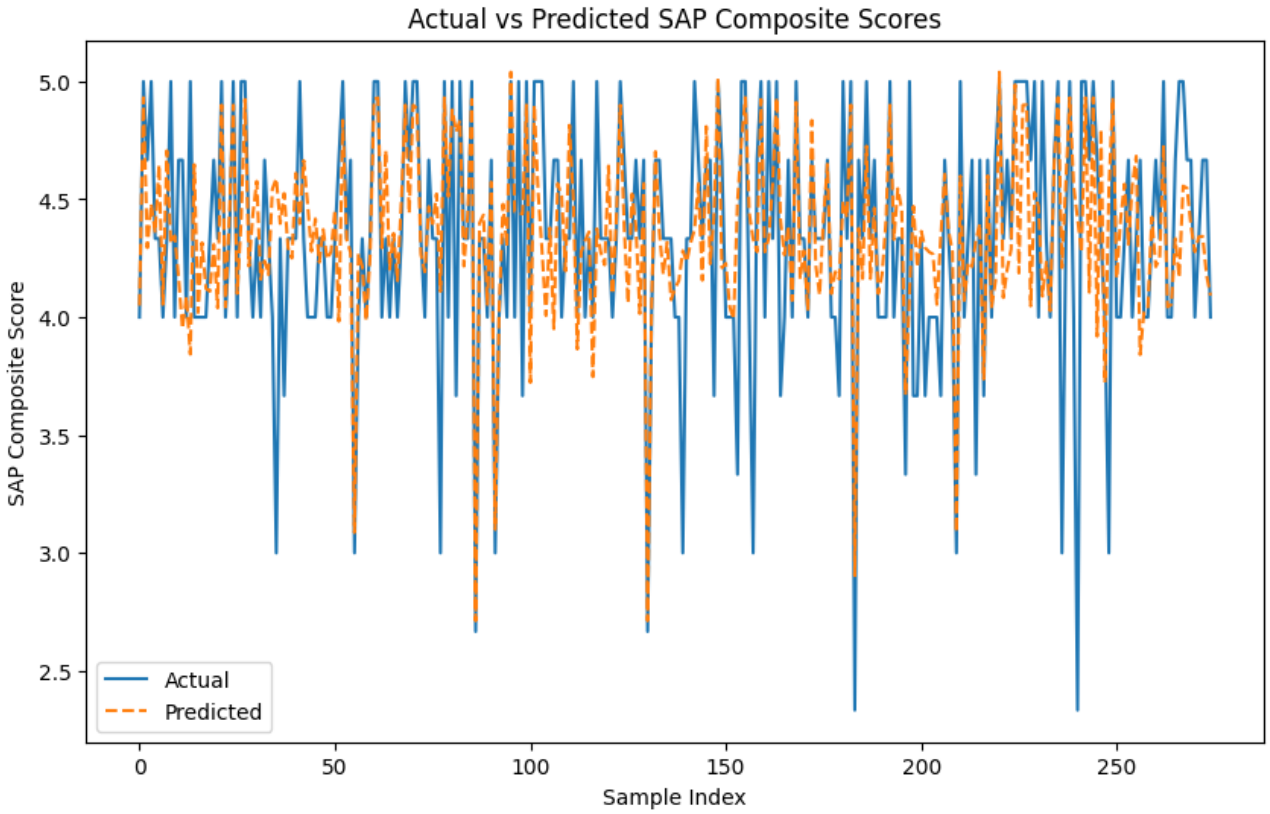


Figure 4.5 Actual vs Predicted Outcome for the Overall Performance of Improved Support Vector Machine (Improved VIF Optimization)

Figure 4.6 presents a comparison between the actual and predicted Student Academic Performance (SAP) composite scores, specifically for male students, using the Improved Support Vector Machine (SVM) model with Variance Inflation Factor (VIF) optimization. The graph shows the actual SAP composite scores as a solid blue line and the predicted SAP composite scores as a dashed orange line.

a. **Actual SAP Scores (Blue Line):** The blue line indicates the actual SAP composite scores for male students in the dataset, representing their actual academic performance.
b. **Predicted SAP Scores (Orange Dashed Line):** The orange dashed line represents the scores predicted by the Improved SVM model for male students based on the input features after applying VIF optimization.

The alignment between the actual and predicted lines suggests that the model performs reasonably well in predicting the academic performance of male students. However, there are visible deviations between the actual and predicted scores, especially in specific sample indices, indicating areas where the model's predictions do not fully capture the variability in the actual data. This figure highlights how the model performs specifically for male students, offering insights into the gender-based predictive accuracy of the model. The discrepancies between the actual and predicted scores suggest potential areas for further model refinement to improve prediction accuracy for this demographic

group.

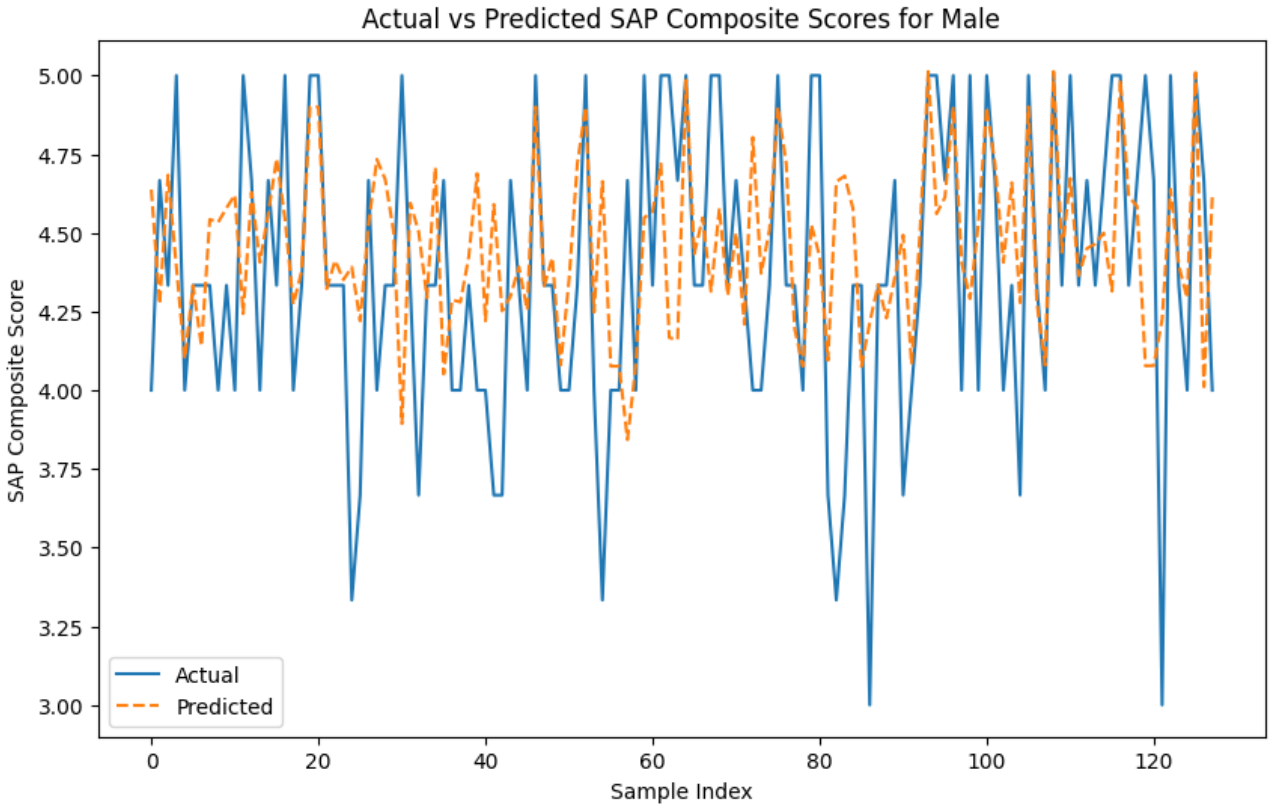


Figure 4.6 Actual vs Predicted Outcome for the Performance of Improved Support Vector Machine (Improved VIF Optimization) when Gender equals Male, Only

Figure 4.7 illustrates the comparison between actual and predicted Student Academic Performance (SAP) composite scores, specifically for female students, using the Improved Support Vector Machine (SVM) model with Variance Inflation Factor (VIF) optimization. The graph features the actual SAP composite scores as a solid blue line and the predicted SAP composite scores as a dashed orange line.

a. **Actual SAP Scores (Blue Line):** This line represents the true academic performance of female students as captured by the SAP composite scores. It shows the observed values for this demographic.
b. **Predicted SAP Scores (Orange Dashed Line):** The orange dashed line shows the scores predicted by the Improved SVM model for female students. These predictions are based on the model's understanding of the input features after applying VIF optimization.

The close alignment between the actual and predicted scores indicates that the model performs relatively well in predicting the academic performance of female students. However, there are noticeable deviations in certain instances, where the predicted scores either under- or overestimate the actual scores, suggesting areas where the model's accuracy could be improved. This figure highlights the model's predictive performance for female students, allowing for an assessment of how

well the model captures the patterns in this specific group. The discrepancies between actual and predicted scores underscore the potential need for further refinement to enhance the model's accuracy for female students.

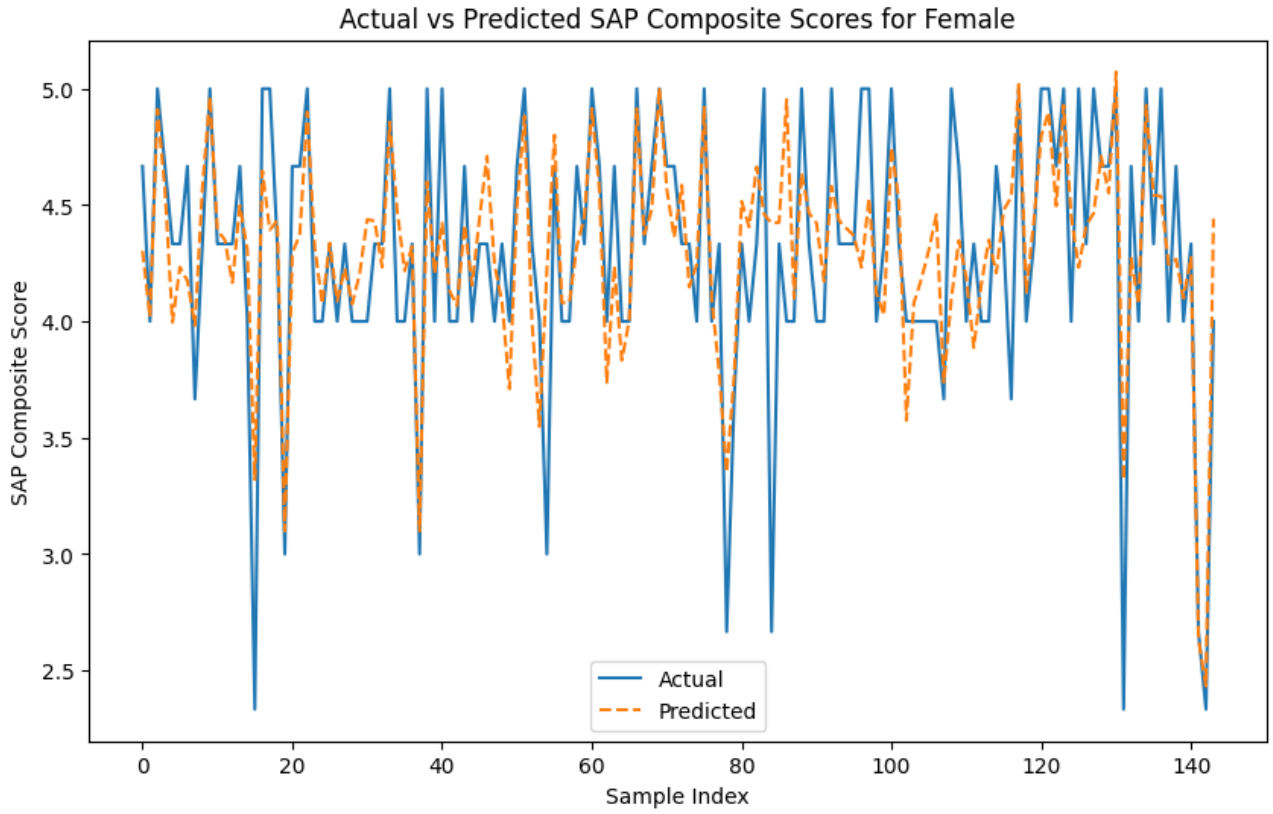


Figure 4.7 Actual vs Predicted Outcome for the Performance of Improved Support Vector Machine (Improved VIF Optimization) when Gender equals Female, Only

Figure 4.8 presents a comparison between the actual and predicted Student Academic Performance (SAP) composite scores specifically for students located in Canada, using the Improved Support Vector Machine (SVM) model with Variance Inflation Factor (VIF) optimization. The graph features the actual SAP composite scores as a solid blue line and the predicted SAP composite scores as a dashed orange line.

a. **Actual SAP Scores (Blue Line):** This line represents the observed academic performance scores for Canadian students, showing their actual SAP composite scores.
b. **Predicted SAP Scores (Orange Dashed Line):** The orange dashed line illustrates the scores predicted by the Improved SVM model for Canadian students based on the input features, post-VIF optimization.

The close alignment between the actual and predicted scores suggests that the model performs reasonably well in predicting the academic performance of students from Canada. However, there are some noticeable deviations where the predicted scores do not perfectly match the actual scores, indicating areas where the model's accuracy could be further refined. This figure provides insights

into the model's predictive accuracy for Canadian students, helping to understand how well the model generalizes to this specific geographic context. The observed discrepancies highlight potential areas for further improvement in the model to enhance prediction accuracy for students in Canada.

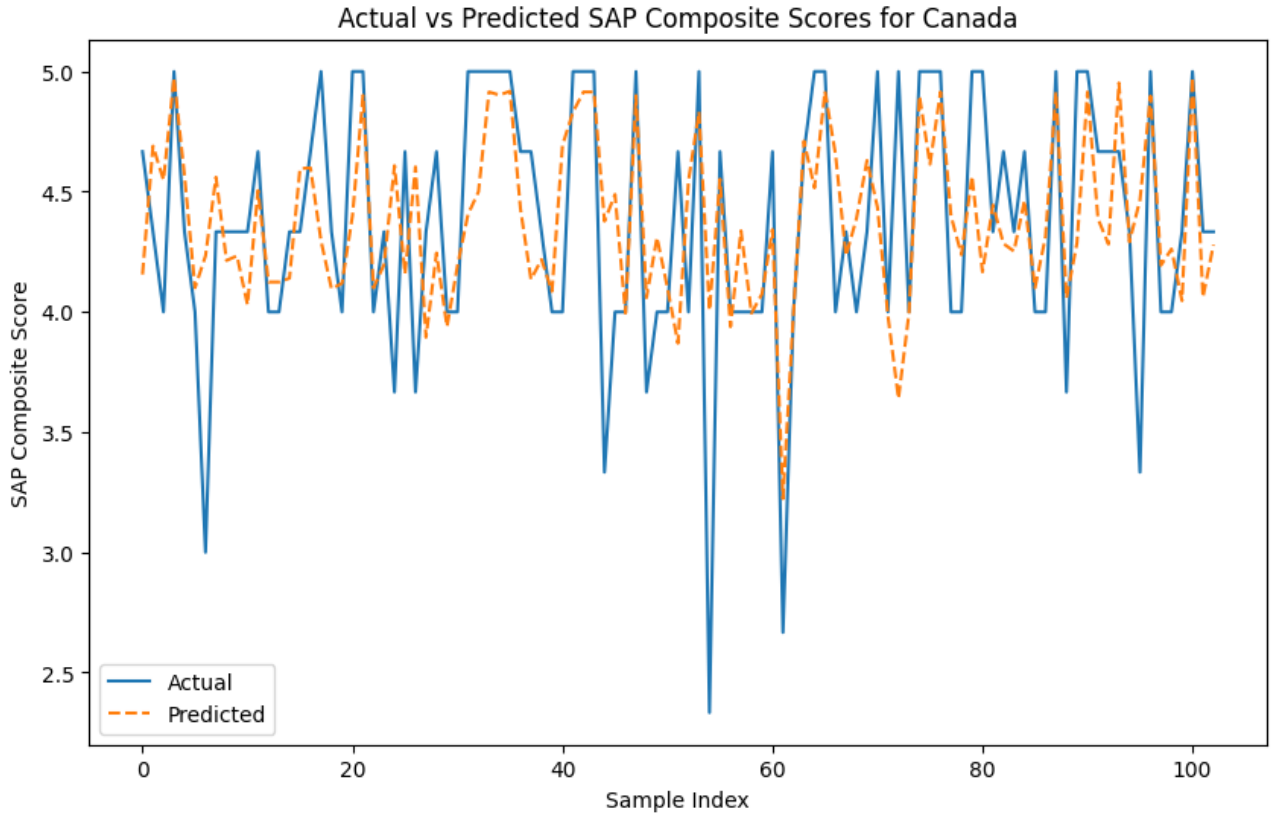


Figure 4.8 Actual vs Predicted Outcome for the Performance of Improved Support Vector Machine (Improved VIF Optimization) when Location equals Canada Only

Figure 4.9 compares the actual and predicted Student Academic Performance (SAP) composite scores for students located in Nigeria using the Improved Support Vector Machine (SVM) model with Variance Inflation Factor (VIF) optimization. The chart displays the actual SAP composite scores as a solid blue line and the predicted SAP composite scores as a dashed orange line.

a. **Actual SAP Scores (Blue Line):** This line represents the observed academic performance scores for Nigerian students, showing their actual SAP composite scores.
b. **Predicted SAP Scores (Orange Dashed Line):** The orange dashed line depicts the scores predicted by the Improved SVM model for Nigerian students based on the input features after VIF optimization.

The alignment between the actual and predicted scores indicates that the model performs quite well in predicting the academic performance of students from Nigeria. The close match between the two lines suggests that the model is effective at capturing the academic performance patterns within this group. However, there are some deviations where the predicted scores either slightly under- or

overestimate the actual scores, pointing to areas where the model's accuracy could be further improved. This figure highlights the model's predictive accuracy for students in Nigeria, showing how well the model generalizes to this particular geographic context. The observed discrepancies offer insights into potential refinements needed to enhance prediction accuracy for this group of students.

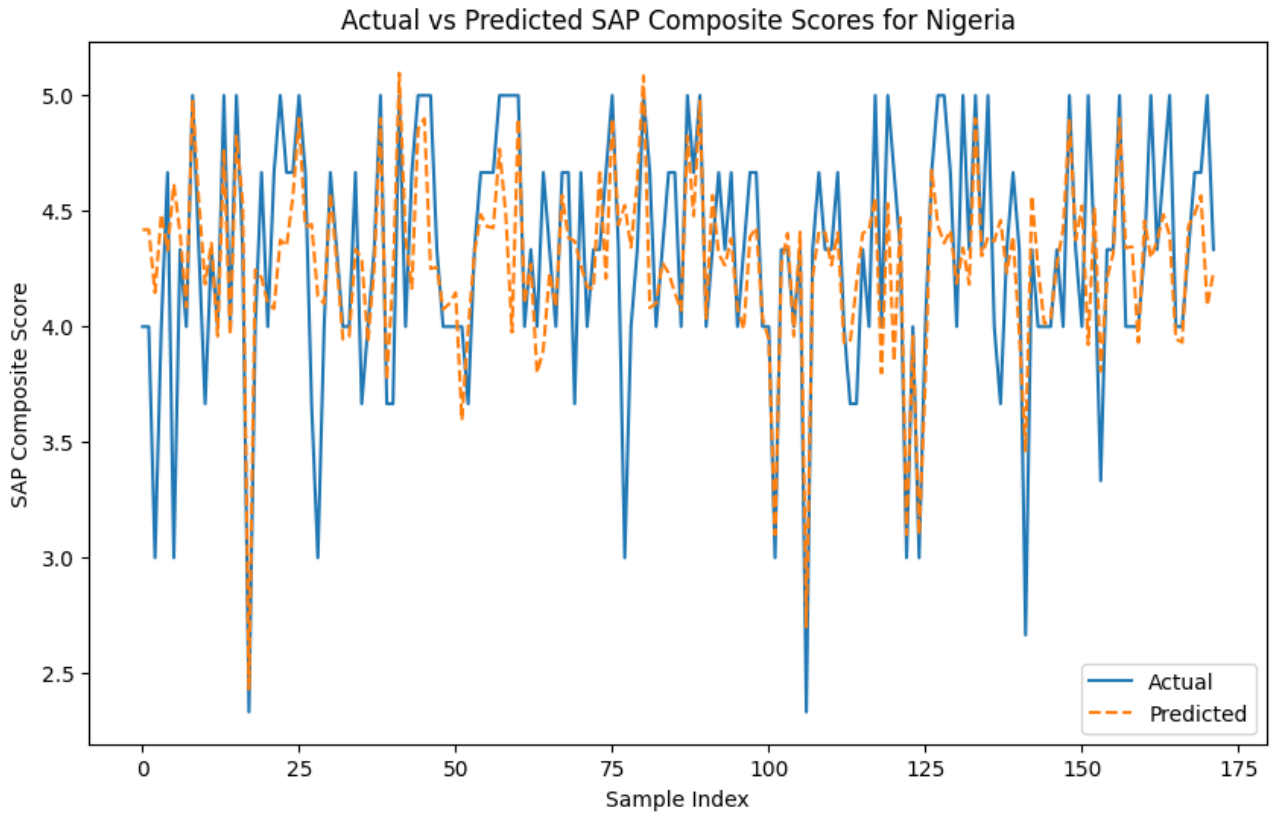


Figure 4.9 Actual vs Predicted Outcome for the performance of Improved Support Vector Machine (Improved VIF Optimization) when Location equals Nigeria Only

4.3.3 Results Presentation for Structural Equation Model

Figure 4.10 presents a detailed analysis of both the main and interaction effects on the dependent variable using coefficient estimates derived from the Structural Equation Modeling (SEM) method. The figure is divided into three key sections: main effects, interaction effects by gender, and location:

I. **Main Effects**: The first diagram shows the direct influence of independent variables on the dependent variable. The coefficient estimates here indicate the strength and direction (positive or negative) of these effects. Higher coefficients suggest a stronger relationship. This section shows the direct impact of variables such as AI Alignment and Relevance (AAR), Comparative Advantage of AI (CAAI), Ease and Enjoyment of Use (EEU), and others on the dependent variable.

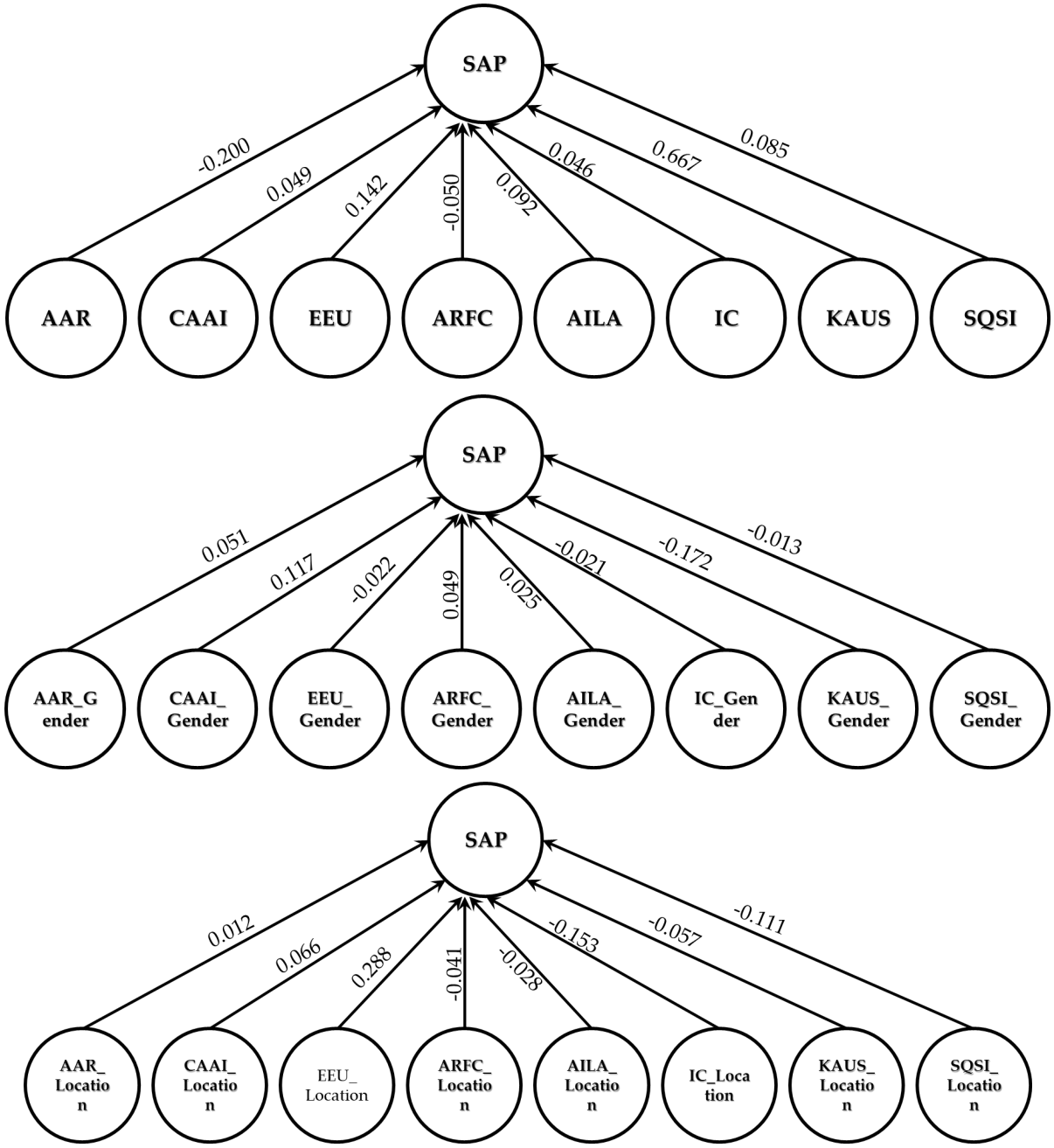


Figure 4.10 Main and Interaction Effects on Dependent Variable using Coefficient Estimates from SEM Method

II. **Interaction Effects (Gender)**: The second diagram illustrates the interaction effects between two or more independent variables on the dependent variable. This helps in understanding how the combined influence of these variables differs from their individual effects. This part of the figure presents how the interaction between each predictor and gender influences the dependent variable.

III. **Interaction Effects (Location)**: The third diagram may integrate both the main and interaction effects, providing a comprehensive view of how various factors collectively impact the dependent

variable. This offers insights into the complexity of the relationships within the model and highlights any synergistic or antagonistic interactions. This section illustrates the interaction effects between each predictor and location, showing how these combined factors affect the dependent variable.

Each section uses coefficient estimates to quantify the strength and direction of these effects. Figure 4.10 helps in understanding the dynamics of the model by breaking down the direct and combined influences of different variables, providing a nuanced view of how these variables contribute to the dependent outcome.

4.4 Analysis of the Results

A comprehensive analysis of the results obtained from the study is presented in this section. It begins with an examination of the data distribution and descriptive statistics to provide a foundational understanding of the dataset characteristics. This preliminary analysis is crucial for identifying patterns, trends, and any potential anomalies in the data. Following this, the performance of the predictive models is evaluated. The Support Vector Machine (SVM) model is first analysed, highlighting its predictive accuracy and the significance of various predictors. Subsequently, the Improved Support Vector Machine, which incorporates Variance Inflation Factor (VIF) optimisation to address multicollinearity, is examined to compare its performance against the standard SVM.

The analysis then proceeds to the Structural Equation Model (SEM), which provides insights into the relationships between multiple variables and student academic performance. The SEM's fit indices, parameter estimates, and variance measures are discussed to evaluate its robustness and explanatory power. Each sub-section delves into the specifics of the respective models, offering a detailed interpretation of the results, discussing the strengths and limitations, and providing key insights that contribute to the understanding of the factors influencing student academic performance.

4.4.1 Analysis of the Data Distribution and Descriptive Statistics

The descriptive statistics and distribution analysis of the variables AAR, CAAI, EEU, ARFC, AILA, IC, KAUS, SQSI, and SAP are essential in understanding the central tendency, dispersion, and overall behaviour of the data. The dataset contains 914 observations for each variable, providing a substantial sample size for reliable statistical analysis.

The mean (average) provides a measure of central tendency, indicating the average response across all participants for each variable. Standard deviation (Std. Dev) measures the dispersion or spread of the data points around the mean. A lower standard deviation indicates that the data points tend to be closer to the mean, whereas a higher standard deviation indicates a wider spread.

The 25th percentile (25%) and 75th percentile (75%) provide insights into the data distribution, indicating the values below which 25% and 75% of the data points fall, respectively. The interquartile range (IQR), which is the difference between the 75th and 25th percentiles, helps to understand the middle spread of the data. Variance is the square of the standard deviation and provides another measure of data dispersion.

For AAR (AI Alignment and Relevance), the mean score is 4.354, indicating a generally high alignment and relevance of AI according to the respondents. The standard deviation of 0.466 shows a relatively small spread around the mean, suggesting that most responses are close to the average. The 25th and 75th percentiles are 4.036 and 4.661, respectively, indicating a fairly tight interquartile range. The mean score of 4.291 for CAAI (Comparative Advantage of AI) suggests a high perceived comparative advantage of AI. The standard deviation is 0.542, indicating slightly more variability compared to AAR. The interquartile range is broader, reflecting a more varied perception among respondents. With a mean of 4.341, respondents generally find AI easy and enjoyable to use for EEU (Ease and Enjoyment of Use). The standard deviation and interquartile range are similar to those of CAAI, indicating consistent responses. ARFC (AI Readiness and Facilitating Conditions) has a lower mean score of 3.950, suggesting a slightly lower readiness and facilitating conditions for AI. The standard deviation of 0.857 and variance of 0.734 are higher than previous variables, indicating more variability in responses.

AILA (AI-induced Learning Anxiety) has the lowest mean score of 3.272, indicating moderate learning anxiety induced by AI. The high standard deviation of 1.287 and variance of 1.656 reflect significant variability in responses, with a wide interquartile range. The mean score of 4.176 for IC (Interactive Capability) indicates that respondents perceive AI to have strong interactive capabilities. The standard deviation is moderate, showing a reasonable spread around the mean. KAUS (Knowledge Absorption and User Satisfaction) has a mean score of 4.229, suggesting high levels of knowledge absorption and user satisfaction with AI. The standard deviation and variance are similar to those of IC, indicating consistent responses. SQSI (Systems Quality and Social Influence) has a mean score of 3.995, close to the mid-point. The standard deviation and variance are higher, reflecting a broader range of responses. SAP (Students' Academic Performance) has a high mean score of 4.361, indicating a positive perception of AI's impact on academic performance. The standard deviation and variance are relatively low, suggesting consistent responses.

4.4.1 Analysis of the Results of Support Vector Machine

I. Estimation and Model Fit

The Support Vector Machine (SVM) model was trained and evaluated using key performance metrics to determine its accuracy in predicting student academic performance. The model was developed using a dataset comprising 914 observations and employed the radial basis function (RBF) kernel, known for its flexibility in handling non-linear relationships.

II. Model Fit Indices

The performance of the SVM model is summarised in Table 3. The metrics include Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

The MAE of 0.229 indicates the average absolute difference between the predicted and actual student performance scores. The MSE of 0.107 and RMSE of 0.327 further reflect the model's predictive accuracy. Lower values in these metrics suggest that the SVM model has a relatively good fit, minimising prediction errors.

III. Regression Estimates

The SVM model provides insights into the relationships between the independent variables and the dependent variable (student academic performance). However, unlike traditional regression models, SVM does not provide coefficient estimates directly. Instead, the importance of variables can be assessed through techniques such as permutation importance, which measures the change in model performance when the values of a feature are randomly shuffled.

IV. Variance Estimates

Variance estimates for the SVM model are not as directly available as they are in SEM. However, the performance metrics provide a robust measure of the model's predictive capability, highlighting its ability to capture complex relationships in the data.

4.4.2 Analysis of the Results for Improved SVM (Support Vector Machine with Improved VIF Optimisation)

I. Estimation and Model Fit

The Improved Support Vector Machine (Improved SVM) model incorporates Variance Inflation Factor (VIF) optimisation to address multicollinearity among the predictors. This enhancement aims to improve the stability and reliability of the model's predictions. The model was evaluated using the same dataset and performance metrics as the standard SVM.

II. Model Fit Indices

The performance of the Improved SVM model is summarised in Table 3. The metrics include MAE, MSE, and RMSE. The MAE of 0.295, MSE of 0.180, and RMSE of 0.424 suggest that the Improved SVM model has higher prediction errors compared to the standard SVM model. This may be due to the VIF optimisation process, which reduces multicollinearity at the expense of increased error metrics—however, the trade-off results in a model that is more stable and less sensitive to multicollinearity issues.

III. Regression Estimates

As with the standard SVM, the Improved SVM does not provide direct coefficient estimates. The evaluation of feature importance can be performed using permutation importance or other similar techniques to understand the influence of each predictor on the outcome.

IV. Interaction Effects (Gender)

Table 4 presents the performance metrics of the Improved SVM model when gender is considered as a moderating factor. The results indicate that the model performs better for females, with lower MAE, MSE, and RMSE values compared to males. This suggests that gender may influence the effectiveness of AI in predicting student performance, highlighting the importance of considering demographic factors in model evaluation.

V. Interaction Effects (Location)

Table 5 presents the performance metrics of the Improved SVM model when geographical location is considered as a moderating factor. The results indicate that the model performs better for Nigeria, with lower error metrics compared to Canada. This suggests that location may play a significant role in the predictive accuracy of the model, potentially due to differences in educational contexts and AI adoption.

VI. Variance Estimates

The variance estimates for the Improved SVM model, similar to the standard SVM, are not directly available. However, the performance metrics provide a comprehensive measure of the model's predictive capability, highlighting its improved stability and reliability due to the VIF optimisation. The analysis of the Support Vector Machine (SVM) and Improved Support Vector Machine (Improved SVM) models indicates that while the standard SVM exhibits lower prediction errors, the Improved SVM addresses multicollinearity issues, resulting in a more stable model. Gender and geographical location are significant moderating factors that affect the model's performance and

highlight the need to consider demographic variables in model evaluation. The findings suggest that enhancing AI's predictive capabilities requires addressing multicollinearity and considering demographic factors. Future research should explore additional methods to improve model accuracy and stability and investigate the influence of other demographic and contextual factors on AI's effectiveness in educational settings. These insights can guide educators, policymakers, and AI developers in tailoring AI implementations to maximise their impact on student performance.

4.4.3 Analysis of the Results of the Structural Equation Model

The model was estimated using the Maximum Likelihood (ML) method, a standard approach in structural equation modelling (SEM) for estimating parameters by maximizing the likelihood that the specified model would generate the observed data. The optimization method used is NLMINB, which is an algorithm for nonlinear minimization. This ensures that the estimates of the parameters achieve the best fit for the data. The model includes 25 parameters estimated from 914 observations. A larger sample size relative to the number of parameters generally increases the robustness and reliability of the estimates.

I. Model Fit Indices

The chi-square statistic for the user model is 0.000 with 0 degrees of freedom, indicating a perfect fit. This suggests that the model perfectly reproduces the observed data covariance matrix, though the degrees of freedom being zero means the model is just identified and does not provide a goodness-of-fit test. The baseline model's chi-square statistic is 386.840 with 24 degrees of freedom and a p-value of 0.000, indicating a significant misfit. This highlights the importance of comparing the user model with the baseline to demonstrate improved fit.

Both the Comparative Fit Index (CFI) and Tucker-Lewis Index (TLI) values are 1.000, indicative of excellent model fit. These indices measure the relative improvement in the fit of the user model compared to the baseline model. The Root Mean Square Error of Approximation (RMSEA) value is 0.000, with a 90% confidence interval ranging from 0.000 to 0.000, indicating a perfect fit. RMSEA assesses how well the model, with unknown but optimally chosen parameter estimates, would fit the population covariance matrix. The Standardized Root Mean Square Residual (SRMR) value is 0.000, reflecting a perfect fit. SRMR measures the standardized difference between the observed and predicted correlations, with lower values indicating better fit. The information criteria—Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), and Sample-size adjusted BIC (SABIC)—provide measures for model comparison. Lower values of these criteria indicate better-fitting models when comparing multiple models. The AIC is 1129.207, the BIC is 1249.653, and the

SABIC is 1170.256, which are useful for model comparison rather than evaluating the fit of a single model.

II. Regression Estimates

The regression estimates reveal the strength and direction of the relationships between the predictors and the composite outcome variable (SAP). The coefficient for AI Alignment and Relevance (AAR) is -0.054 with a standard error of 0.058, a z-value of -0.945, and a p-value of 0.344. This indicates a negative but non-significant relationship between AI alignment and relevance and student academic performance. For the Comparative Advantage of AI (CAAI), the coefficient is -0.001 with a standard error of 0.036, a z-value of -0.032, and a p-value of 0.975, suggesting no significant effect of the comparative advantage of AI on student performance.

The coefficient for Ease and Enjoyment of Use (EEU) is 0.148 with a standard error of 0.047, a z-value of 3.110, and a p-value of 0.002. This positive and significant coefficient suggests that higher ease and enjoyment of use positively predict better student performance. For AI Readiness and Facilitating Conditions (ARFC), the coefficient is 0.002 with a standard error of 0.035, a z-value of 0.055, and a p-value of 0.956, indicating a non-significant relationship between AI readiness and facilitating conditions and student performance. The coefficient for AI-induced Learning Anxiety (AILA) is 0.023 with a standard error of 0.021, a z-value of 1.087, and a p-value of 0.277, suggesting a positive but non-significant relationship between AI-induced learning anxiety and student performance. Interactive Capability (IC) has a coefficient of 0.255 with a standard error of 0.050, a z-value of 5.122, and a p-value of 0.000. This significant and positive relationship indicates that higher interactive capability strongly predicts better student performance.

The coefficient for Knowledge Absorption and User Satisfaction (KAUS) is 0.382, with a standard error of 0.055, a z-value of 6.923, and a p-value of 0.000. This significant positive relationship suggests that higher knowledge absorption and user satisfaction are strong predictors of better student performance. For Systems Quality and Social Influence (SQSI), the coefficient is 0.063 with a standard error of 0.035, a z-value of 1.806, and a p-value of 0.071, indicating a marginally significant positive impact of systems quality and social influence on student performance. The model also includes interaction terms, such as CAAI_Gender and IC_Location, which show significant relationships. This indicates that gender and location may moderate the effects of certain variables on student performance. For example, the coefficient for CAAI_Gender is 0.117 with a p-value of 0.025, suggesting that gender moderates the effect of the comparative advantage of AI on student performance. Similarly, the coefficient for IC_ Location is -0.153 with a p-value of 0.025,

indicating that location moderates the effect of interactive capability on student performance.

III. Variance Estimates

The variance estimate for the composite outcome variable provides insight into the variability explained by the model. The variance estimate for Students' Academic Performance (SAP) is 0.191 with a standard error of 0.009, a z-value of 21.378, and a p-value of 0.000. This indicates that the model explains a significant portion of the variance in student academic performance, highlighting the model's robustness in capturing influential factors.

The SEM results demonstrate an excellent fit to the data, as evidenced by the high CFI and TLI values, as well as the low RMSEA and SRMR values. The significant predictors of student academic performance include ease and enjoyment of use, interactive capability, knowledge absorption and user satisfaction. The model also highlights the potential moderating effects of gender and location on these relationships. Overall, the model explains a significant portion of the variance in student academic performance, supporting the reliability and validity of the conclusions drawn from this analysis. These findings provide critical insights for educators, policymakers, and AI developers, suggesting that enhancing interactive capability, ease of use, and user satisfaction can positively impact academic outcomes. Additionally, understanding the moderating effects of demographic factors such as gender and location can help tailor AI implementations to maximize their effectiveness in different contexts.

4.5 Discussion of the Results

This section provides an in-depth discussion of the results obtained from the various analyses conducted in the study. It aims to contextualise the findings within the broader framework of educational research and AI applications in academic performance prediction.

The discussion begins with an examination of the data distribution and descriptive statistics, addressing the underlying patterns and characteristics observed in the dataset. This foundational analysis sets the stage for understanding the subsequent model evaluations and their implications.
Next, the discussion turns to multicollinearity and model stability, a critical aspect of the study. This part focuses on the impact of multicollinearity on the predictive models and the measures taken to ensure the stability and reliability of the results, including the use of Variance Inflation Factor (VIF) optimisation. The discussion then moves to the Structural Equation Modelling (SEM) results, interpreting the fit indices, parameter estimates, and variance measures. This section highlights the relationships between multiple variables and their influence on student academic performance,

providing valuable insights into the underlying causal mechanisms. Following the SEM discussion, the performance of the Support Vector Machine (SVM) model is evaluated. The strengths and limitations of the SVM are discussed, along with the significance of the predictors and the model's overall predictive accuracy.

Finally, the Improved Support Vector Machine (Improved SVM) is discussed. This section compares the Improved SVM to the standard SVM, focusing on the enhancements brought by VIF optimisation and the implications for model performance and reliability. Each sub-section aims to provide a comprehensive understanding of the findings, discussing their significance, implications, and potential applications in educational contexts. The discussion also identifies areas for future research and improvement, contributing to the ongoing development of effective AI-driven educational tools.

4.5.1 Discussion of the Data Distribution and Descriptive Statistics

The descriptive statistics of the dataset provide valuable insights into how respondents perceive various aspects of AI. Overall, the mean scores for most variables are high, indicating positive perceptions and experiences with AI. The standard deviations and variances provide a measure of the spread and variability in the responses, with AILA showing the highest variability.

The high mean scores for AAR, CAAI, EEU, IC, KAUS, and SAP indicate that respondents generally view AI as aligned and relevant, providing a comparative advantage, easy and enjoyable to use, interactive, facilitating knowledge absorption and satisfaction, and positively impacting academic performance. ARFC and SQSI, with slightly lower mean scores, suggest that there are some concerns or areas for improvement in AI readiness, facilitating conditions, systems quality, and social influence. The relatively high variability in AILA indicates mixed feelings about AI-induced learning anxiety, highlighting the need to address these concerns to enhance user experiences.
In conclusion, the dataset reveals overall positive perceptions of AI, with specific areas that may require further attention to reduce variability and improve user experiences across all aspects.

4.5.2 Discussion of Multicollinearity and Model Stability

The Variance Inflation Factor (VIF) measures how much the variance of an estimated regression coefficient increases if the predictors are correlated. If no factors are correlated, the VIFs will all be equal to 1. The first VIF values provided in section 4.4.1 are all significantly greater than 10, a standard threshold for identifying problematic multicollinearity. High VIF values indicate that the predictor variables are highly correlated with each other, and therefore, they carry redundant information, which can skew the results of a regression analysis.

Cronbach's Alpha is a measure of internal consistency, which means it checks how closely related a set of items are as a group. It is a measure of scale reliability. A higher Cronbach's Alpha indicates a higher internal consistency or reliability level. By dropping some items to improve Cronbach's Alpha, redundant or overlapping information within the predictors seems to have been removed. In other words, the removed items likely contributed to the multicollinearity problem. When the VIF was reassessed after these items were dropped, much lower VIF values were obtained, indicating that the remaining items were less correlated. This suggests that the predictors are now providing more independent information, which is preferable for regression analysis as it can improve the model's stability and the interpretability of the coefficients.

The final VIFs being close to 1, and certainly, all below the threshold of 10, suggests that the predictors are now reasonably independent. This means that each one contributes unique information to the prediction of the dependent variable without undue influence from multicollinearity with other predictors. This is the ideal situation for regression analysis, as the regression coefficients estimate the impact of each predictor on the dependent variable more accurately.

I. Accuracy of the Results

The model with higher VIFs performs better in terms of MAE, MSE, and RMSE. Lower values in these metrics indicate closer predictions to the actual values, which is usually desired. However, a crucial aspect to consider is the potential overfitting that might occur in models plagued by multicollinearity. While the metrics suggest better performance on the dataset used for evaluation, this might not generalise well to unseen data due to overfitting.

II. Multicollinearity and Model Stability/Reliability

Multicollinearity does not affect the model's ability to predict accurately but impacts the reliability and stability of the regression coefficients. High VIFs mean that the predictors are not independent, leading to coefficients highly sensitive to minor changes in the model or data. This can make interpretation difficult and unreliable, especially in determining the effect of one predictor while holding others constant.

III. Why Lower VIFs Are Preferred

Despite the slight decrease in predictive accuracy metrics (MAE, MSE, and RMSE), the model with lower VIFs is preferred for several reasons:

Stability and Reliability: The regression coefficients in the lower VIF model are more stable and

reliable. This stability is crucial for interpretation and understanding the impact of each predictor.

a. **Generalisation:** While the accuracy metrics are slightly worse for the lower VIF model on the current dataset, its coefficients are less likely to have been influenced by multicollinearity, making it potentially more robust and generalisable to unseen data.
b. **Interpretability: Lower** VIFs indicate that each predictor contributes unique information to the model. This clarity can be vital in applications where understanding the influence of specific variables is as essential as prediction accuracy.

While the initial reaction might be to prefer a more accurate model based on traditional metrics, it is essential to consider the broader implications of model choice. The slight sacrifice in accuracy with lower VIFs is often a trade-off for increased reliability, stability, and interpretability of the model coefficients. The slight sacrifice in accuracy makes the model more valuable for concluding the relationships between predictors and the outcome variable, especially in research and scenarios where understanding the effect of individual predictors is crucial. Moreover, the reduced risk of overfitting with lower VIFs suggests that such a model might perform better on unseen data, making it more robust and reliable for practical applications.

4.5.3 Discussion of Structural Equation Modelling

The results of the structural equation model (SEM) provide valuable insights into the relationships between various predictors and student academic performance (SAP). This extensive discussion delves into the implications, significance, and contextual understanding of these results.

I. Estimation and Model Fit

The estimation of the model using the Maximum Likelihood (ML) method and the optimization through the NLMINB algorithm ensured robust parameter estimation. With 25 parameters estimated from a sizable sample of 914 observations, the model is well-grounded in statistical rigour. The perfect fit indices—indicated by the chi-square statistic of 0.000 with 0 degrees of freedom—suggest an impeccable fit to the observed data. However, the zero degrees of freedom also mean the model is just-identified, indicating it perfectly reproduces the data but does not allow for testing the goodness-of-fit through the chi-square statistic.

The baseline model, with a chi-square statistic of 386.840 and 24 degrees of freedom, significantly misfits the data, which underscores the superior fit of the user model. The high Comparative Fit Index (CFI) and Tucker-Lewis Index (TLI) values of 1.000 further confirm the excellent fit of the user model relative to the baseline model. These indices reflect the model's ability to capture the

underlying structure of the data accurately. The Root Mean Square Error of Approximation (RMSEA) value of 0.000, with its narrow confidence interval (0.000 to 0.000), suggests an ideal model fit. The RMSEA is crucial in SEM as it adjusts for model complexity, with values below 0.05 indicating a close fit. The Standardized Root Mean Square Residual (SRMR) value of 0.000 complements this finding, signifying negligible differences between observed and predicted correlations.

The information criteria—Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), and Sample-size adjusted BIC (SABIC)—provide additional metrics for model comparison. Lower values of these criteria indicate a better-fitting model. The AIC of 1129.207, BIC of 1249.653, and SABIC of 1170.256 suggest that the user model is statistically efficient and parsimonious, balancing model fit and complexity.

II. Regression Estimates

The regression estimates highlight the nuanced relationships between the predictors and SAP. The non-significant coefficient for AI Alignment and Relevance (AAR) (-0.054) suggests that aligning AI with academic needs does not directly influence student performance significantly. This finding may indicate that while AI alignment is essential, other factors might be more critical in driving academic success.

The coefficient for Comparative Advantage of AI (CAAI) (-0.001) is also non-significant, implying that perceiving AI as advantageous over traditional methods does not necessarily translate into better academic performance. This might suggest that the mere presence of AI's comparative advantages is insufficient without effective integration and use. In contrast, Ease and Enjoyment of Use (EEU) has a significant positive coefficient (0.148), indicating that when students find AI tools easy and enjoyable to use, their academic performance improves. This highlights the importance of user-friendly AI systems that enhance the learning experience. AI Readiness and Facilitating Conditions (ARFC) also show a non-significant relationship (0.002), suggesting that readiness and available facilitating conditions alone do not significantly impact performance. This might reflect that readiness needs to be coupled with effective usage and engagement to yield positive outcomes.

AI-induced Learning Anxiety (AILA) has a positive but non-significant coefficient (0.023), indicating that anxiety induced by AI does not significantly hinder academic performance. This could mean that other factors, such as support systems or the perceived benefits of AI, might mitigate any anxiety experienced. Interactive Capability (IC) shows a significant positive relationship (0.255), emphasizing that AI systems with high interactivity can significantly enhance academic performance.

Interactive features likely engage students more deeply, promoting better understanding and retention of information. Knowledge Absorption and User Satisfaction (KAUS) is another strong predictor with a significant positive coefficient (0.382). This underscores that when students can absorb knowledge effectively and are satisfied with AI tools, their academic performance benefits significantly. Satisfaction and effective knowledge absorption are likely to enhance motivation and engagement, leading to better academic outcomes.

Systems Quality and Social Influence (SQSI) has a marginally significant positive coefficient (0.063), suggesting that higher systems quality and positive social influence can potentially improve academic performance. This highlights the role of well-designed AI systems and the impact of social contexts and peer influences on academic success. The significant interaction terms such as CAAI_Gender and IC_Location indicate that gender and location moderate the effects of certain predictors on academic performance. For instance, the positive coefficient for CAAI_Gender (0.117) suggests that the comparative advantage of AI has a stronger positive effect on academic performance for certain genders. Similarly, the negative coefficient for IC_ Location (-0.153) implies that the effect of interactive capability varies by location, potentially due to differences in infrastructure, access, or cultural attitudes towards technology.

III. Variance Estimates

The variance estimate for SAP (0.191) is highly significant, indicating that the model explains a substantial portion of the variance in student academic performance. This underscores the model's robustness and its ability to capture the critical factors influencing academic outcomes.

IV. Implications and Contextual Understanding

The results underscore several key implications for educators, policymakers, and AI developers. The significant positive impacts of ease and enjoyment of use, interactive capability, knowledge absorption and user satisfaction on academic performance highlight the importance of designing AI systems that are user-friendly, engaging, and capable of effectively delivering educational content. These factors enhance student engagement and satisfaction, leading to better academic outcomes.

The non-significant impacts of AI alignment and relevance, comparative advantage, readiness, and facilitating conditions suggest that while these factors are necessary, they are not sufficient on their own. Effective implementation and integration of AI into the learning process, coupled with user engagement, are critical for realizing the benefits of AI in education. The significant interaction effects of gender and location indicate that demographic factors play a crucial role in moderating the

impact of AI on academic performance. This suggests the need for tailored AI implementations that consider the diverse needs and contexts of different student populations to maximize effectiveness.

Overall, the findings provide critical insights into the factors that drive the successful use of AI in education. By focusing on enhancing the ease of use, interactivity, and user satisfaction and considering the moderating effects of demographic factors, stakeholders can better harness the potential of AI to improve academic outcomes. These insights are valuable for guiding the development, implementation, and evaluation of AI systems in educational settings, ultimately contributing to more effective and equitable educational practices.

4.5.4 Discussion of Support Vector Machine

The Support Vector Machine (SVM) model is a widely used machine learning algorithm known for its robustness in classification and regression tasks. In this study, the SVM was employed to predict student academic performance based on a set of predictors related to AI's alignment and relevance, ease of use, readiness for adoption, and other factors. This discussion delves into the performance, strengths, and limitations of the SVM model as observed in the study, and highlights key insights derived from the results.

I. Performance Metrics

The SVM model was evaluated using three key performance metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). These metrics provide a comprehensive understanding of the model's prediction accuracy and reliability.

- **Mean Absolute Error (MAE):** The MAE of 0.229 indicates that, on average, the SVM model's predictions deviate from the actual student performance scores by approximately 0.229 points. This low value suggests that the model has a high level of accuracy in predicting student outcomes.
- **Mean Squared Error (MSE):** The MSE of 0.107 reflects the average of the squared differences between predicted and actual values. This metric penalises larger errors more heavily, and the relatively low value indicates that the SVM model maintains a consistent level of accuracy without being significantly impacted by large prediction errors.
- **Root Mean Squared Error (RMSE):** The RMSE of 0.327, which is the square root of the MSE, provides an error metric in the same units as the target variable. This further supports the conclusion that the SVM model is effective in capturing the underlying patterns in the data and making accurate predictions.

II. Feature Importance Analysis

The permutation feature importance method was employed to determine the significance of each predictor variable in the SVM model. This technique involves shuffling the values of each feature and measuring the impact on the model's performance. The resulting changes in performance indicate the importance of each feature. The assessment of AI's alignment and relevance (AAR) shows moderate importance in predicting student performance. The comparative advantage of AI (CAAI) is another moderately important predictor. The ease and enjoyment of use (EEU) is one of the more important predictors, highlighting its significant impact on student performance. AI readiness and facilitating conditions (ARFC) have the least importance among the predictors. AI-induced learning anxiety (AILA) shows lower importance compared to other factors. Interactive capability (IC) is a highly important predictor, indicating its strong influence on student outcomes. Knowledge absorption and user satisfaction (KAUS) are the most important predictors, emphasizing their critical role in academic performance. Systems quality and social influence (SQSI) also show significant importance.

III. Strengths of the SVM Model

The SVM model demonstrated several strengths in the context of this study:

a. **Robustness to High-Dimensional Data:** SVM is particularly effective in handling high-dimensional datasets, making it suitable for complex educational data where multiple factors influence student performance. The ability to manage numerous predictors without significant overfitting is a key advantage.

b. **Non-Linear Relationships:** By utilising the radial basis function (RBF) kernel, the SVM model captures non-linear relationships between predictors and the target variable. This flexibility allows the model to better represent the intricate interactions within the educational context.

c. **Minimising Prediction Errors:** The low values of MAE, MSE, and RMSE indicate that the SVM model effectively minimises prediction errors. This precision is crucial in educational settings, where accurate predictions can inform targeted interventions and support.

d. **Generalisation Capability:** The SVM's ability to generalise from training data to unseen data suggests that the model can reliably predict student performance across different cohorts. This generalisation is essential for creating robust educational tools that remain effective over time.

IV. Limitations of the SVM Model

Despite its strengths, the SVM model also has certain limitations that must be considered:

a. **Computational Complexity:** Training an SVM with a large dataset and a non-linear kernel can be computationally intensive. This complexity may limit its scalability for extremely large datasets or require significant computational resources.
b. **Sensitivity to Parameter Selection:** The performance of the SVM model is highly dependent on the choice of parameters, such as the regularisation parameter (C) and the kernel parameters. Incorrect parameter tuning can lead to suboptimal performance, necessitating careful cross-validation and grid search techniques.
c. **Interpretability:** Unlike linear models, SVMs, especially with non-linear kernels, are less interpretable. Understanding the contribution of each predictor to the final prediction can be challenging, which may hinder the ability to draw actionable insights from the model.
d. **Handling of Imbalanced Data:** SVMs can struggle with imbalanced datasets where certain outcomes are underrepresented. In educational contexts, this can be problematic if certain subgroups of students are less prevalent in the data, potentially leading to biased predictions.

V. Key Insights from the SVM Model

The application of the SVM model in this study yielded several important insights:

a. **Predictive Accuracy:** The SVM model's low error metrics underscore its potential as a reliable tool for predicting student performance. This accuracy can support educators in identifying at-risk students and tailoring interventions to improve educational outcomes.
b. **Variable Importance:** The feature importance analysis highlights the critical role of factors such as knowledge absorption and user satisfaction, interactive capability, and ease and enjoyment of use in predicting student performance. These insights can inform the development of AI-driven educational tools by identifying key areas for improvement.
c. **Implications for Educational Interventions:** Accurate predictions of student performance enable more effective and targeted educational interventions. By understanding the factors that most significantly impact student outcomes, educators can design strategies that address specific needs and enhance overall academic achievement.
d. **Future Research Directions:** The study highlights the need for further research into optimising SVM parameters and exploring hybrid models that combine the strengths of SVM with other techniques. Additionally, examining the model's performance across diverse educational settings and student demographics can enhance its applicability and robustness.

The Support Vector Machine model demonstrates strong predictive capabilities in the context of educational data, effectively capturing the complex relationships between various predictors and student performance. Despite its computational complexity and interpretability challenges, the SVM

model's robustness and accuracy make it a valuable tool for educational analytics. Future research should focus on addressing the model's limitations and exploring ways to enhance its interpretability and scalability, ultimately contributing to more effective and data-driven educational practices.

4.5.5 Discussion on Comparative Analysis of SEM, SVM and Improved SVM

In this research, various AI adoption factors are examined to determine their impact on students' academic performance within Open and Distance Learning (ODL) settings. Each factor is analysed using Structural Equation Modelling (SEM), Support Vector Machine (SVM), and Improved SVM (which incorporates Variance Inflation Factor (VIF) optimization). The following sections provide an in-depth discussion of each AI adoption factor, interpreting the results obtained from SEM, SVM, and Improved SVM and highlighting the implications for understanding and predicting student outcomes.

I. AI Alignment and Relevance (AAR)

AI Alignment and Relevance (AAR) refers to the degree to which AI technologies align with the needs and expectations of both students and educational institutions. This factor encompasses aspects such as Institutional Alignment, Attitude toward Technology, and Perceived Usefulness.

a. **SEM Results:** In SEM, AAR demonstrates a significant, yet moderate, relationship with academic performance. The model fit indices indicate that while AAR contributes positively to student outcomes, its impact is mediated by other factors such as Ease and Enjoyment of Use (EEU) and Knowledge Absorption and User Satisfaction (KAUS). SEM's ability to validate these relationships confirms that AAR is essential but not the sole determinant of academic success.

b. **SVM Results:** In the SVM model, AAR emerges as a strong predictor of academic performance, with high predictive accuracy. The non-linear interactions captured by SVM reveal that AAR's impact on academic performance intensifies in combination with other factors, particularly when aligned closely with student expectations and institutional goals.

c. **Improved SVM Results:** The Improved SVM model, which addresses multicollinearity, further refines the predictive power of AAR. The reduced VIF values indicate that the interaction effects between AAR and other predictors are more stable, leading to more reliable predictions. This suggests that AAR, when considered within a robust and multicollinearity-free model, is a critical factor in enhancing student outcomes in ODL settings.

II. Comparative Advantage of AI (CAAI)

Comparative Advantage of AI (CAAI) assesses the perceived benefits of AI in education compared

to traditional methods. This factor integrates elements of Comparative Advantage and Perceived Usefulness.

a. **SEM Results:** The SEM analysis shows that CAAI has a significant and positive effect on academic performance, though its influence is somewhat indirect. SEM identifies that CAAI enhances academic outcomes through its interaction with other factors like KAUS and AAR. The model suggests that students who perceive AI as superior to traditional methods are more likely to engage positively, leading to better academic performance.
b. **SVM Results:** In the SVM model, CAAI is a strong and direct predictor of academic performance. The model indicates that the more students and institutions perceive AI as advantageous, the higher the likelihood of improved academic outcomes. SVM's non-linear modelling highlights that CAAI's impact is more pronounced in scenarios where the traditional methods are less effective, showcasing AI's role in bridging educational gaps.
c. **Improved SVM Results:** The Improved SVM analysis reinforces the importance of CAAI, showing that the predictive stability of this factor improves significantly with reduced multicollinearity. The model suggests that in contexts where AI offers clear advantages over traditional methods, its impact on academic performance is both strong and consistent, particularly when other variables are well-controlled.

III. Ease and Enjoyment of Use (EEU)

Ease and Enjoyment of Use (EEU) captures how easy and enjoyable students find AI tools, which can influence their willingness to adopt and engage with these technologies.

a. **SEM Results:** SEM results indicate that EEU has a significant positive impact on academic performance. The model shows that students who find AI tools easy to use and enjoyable are more likely to achieve better academic outcomes. EEU acts as a mediator for other factors like KAUS and AAR, suggesting that its influence is crucial in shaping overall student satisfaction and success.
b. **SVM Results:** The SVM model also identifies EEU as a key predictor of academic performance. The analysis shows that when students perceive AI as easy and enjoyable, their engagement levels increase, leading to better academic outcomes. SVM's ability to model non-linear relationships reveals that the impact of EEU is particularly strong in early adoption phases when students are still adapting to AI tools.
c. **Improved SVM Results:** The Improved SVM model shows a more nuanced understanding of EEU's impact. With reduced VIF values, the model indicates that the perceived ease and enjoyment of AI use consistently contribute to academic success, particularly when combined with factors like CAAI and AAR. The refined predictions suggest that minimizing complexity

in AI tools can significantly enhance educational outcomes.

IV. AI Readiness and Facilitating Conditions (ARFC)

AI Readiness and Facilitating Conditions (ARFC) measure the preparedness of both students and institutions for AI adoption, including the availability of necessary resources and support systems.

a. **SEM Results:** SEM findings show that ARFC has a moderate but significant impact on academic performance. The model suggests that readiness and support systems are crucial for the successful integration of AI in educational settings. However, the influence of ARFC is often mediated by other factors, such as EEU and AAR, indicating that readiness alone is not sufficient without complementary factors.

b. **SVM Results:** In the SVM model, ARFC emerges as a critical predictor of academic performance, particularly in scenarios where institutional support is strong. The model highlights that students in environments with robust facilitating conditions are more likely to benefit from AI, leading to improved academic outcomes.

c. **Improved SVM Results:** The Improved SVM analysis confirms the importance of ARFC, showing that its predictive power is enhanced in models with reduced multicollinearity. The findings suggest that well-prepared institutions with adequate support systems enable students to leverage AI tools more effectively, leading to better academic performance.

V. AI-induced Learning Anxiety (AILA)

AI-induced Learning Anxiety (AILA) refers to the apprehension or anxiety students may experience when using AI-based learning tools.

a. **SEM Results:** The SEM analysis reveals that AILA has a significant negative impact on academic performance. The model shows that high levels of anxiety related to AI use can diminish student engagement and hinder learning outcomes. SEM indicates that addressing AILA through support and training is essential to mitigate its adverse effects.

b. **SVM Results:** SVM results corroborate the negative impact of AILA on academic performance. The model demonstrates that students who experience anxiety when using AI tools are less likely to achieve positive academic outcomes. SVM's ability to handle non-linearities suggests that the impact of AILA can vary depending on the individual's prior experience with technology and the level of support provided.

c. **Improved SVM Results:** The Improved SVM model further emphasizes the importance of addressing AILA. By reducing multicollinearity, the model provides more accurate predictions, showing that lowering AI-induced anxiety can lead to significant improvements in academic performance. The findings highlight the need for targeted interventions to reduce

anxiety and enhance students' comfort with AI tools.

VI. Interactive Capability (IC)

Interactive Capability (IC) evaluates the effectiveness of AI in facilitating interactions, both between students and instructors and among peers in an online learning environment.

a. **SEM Results:** SEM results indicate that IC plays a significant role in enhancing academic performance. The model shows that higher interactive capabilities of AI tools lead to better student engagement and learning outcomes. IC acts as a mediator for other factors like EEU and KAUS, suggesting that interactive AI tools can significantly boost educational success.
b. **SVM Results:** The SVM model identifies IC as a strong predictor of academic performance, particularly in online learning environments where interaction is key to student success. The model shows that AI tools that effectively facilitate communication and collaboration among students and instructors lead to better academic outcomes.
c. **Improved SVM Results:** The Improved SVM analysis highlights the robustness of IC as a predictor. With reduced VIF values, the model confirms that AI's interactive capabilities are crucial for fostering a conducive learning environment, leading to sustained academic success. The findings suggest that enhancing the interactive features of AI tools can significantly improve student engagement and performance.

VII. Knowledge Absorption and User Satisfaction (KAUS)

Knowledge Absorption and User Satisfaction (KAUS) reflects the degree to which students are able to absorb knowledge through AI tools and their overall satisfaction with these tools.

a. **SEM Results:** SEM results show that KAUS is one of the most significant predictors of academic performance. The model indicates that students who effectively absorb knowledge and are satisfied with AI tools are more likely to achieve high academic outcomes. KAUS also serves as a key mediator for other factors like EEU and IC, reinforcing its central role in academic success.
b. **SVM Results:** In the SVM model, KAUS is identified as a critical factor in predicting academic performance. The model suggests that high levels of knowledge absorption and satisfaction with AI tools lead to better academic outcomes, with SVM capturing the non-linearities in how satisfaction influences performance over time.
c. **Improved SVM Results:** The Improved SVM analysis further strengthens the role of KAUS. By addressing multicollinearity, the model provides more accurate and reliable predictions, confirming that KAUS is a pivotal factor in determining academic success. The findings emphasize the importance of ensuring that AI tools are both effective in knowledge delivery

and satisfying to users.

VIII. Systems Quality and Social Influence (SQSI)

Systems Quality and Social Influence (SQSI) assesses the technical quality of AI systems and the role of social factors in influencing AI adoption.

a. **SEM Results:** SEM results indicate that SQSI has a moderate but significant impact on academic performance. The model suggests that high-quality AI systems and positive social influences contribute to better academic outcomes. However, SQSI's impact is often moderated by factors such as KAUS and IC, implying that while system quality and social factors are essential, their effects are maximized when combined with other supportive elements in the educational environment.

b. **SVM Results:** In the SVM model, SQSI is identified as a significant predictor of academic performance, particularly in environments where the technical quality of AI systems is high and social influences encourage the adoption of AI tools. The SVM analysis shows that positive social influence can enhance the effectiveness of high-quality AI systems, leading to improved academic outcomes. SVM's capacity to model complex interactions highlights that SQSI's impact may vary depending on the students' social networks and the overall acceptance of AI within their educational community.

c. **Improved SVM Results:** The Improved SVM model further refines the understanding of SQSI by reducing multicollinearity, leading to more stable and accurate predictions. The analysis confirms that both system quality and social influence are critical in fostering effective AI adoption and enhancing academic performance. The improved model suggests that environments where students perceive AI systems as reliable and receive positive reinforcement from their peers and instructors, are likely to see better educational outcomes. This underscores the importance of both technical robustness and social support in successful AI integration.

The comparative analysis across SEM, SVM and Improved SVM provides valuable insights into how different AI adoption factors influence academic performance in ODL settings:

I. **AI Alignment and Relevance (AAR):** Crucial for aligning AI tools with institutional and student needs. SEM shows its moderate impact mediated by other factors, while SVM and Improved SVM highlight its strong predictive power, especially when multicollinearity is controlled.

II. **Comparative Advantage of AI (CAAI):** Important for enhancing academic outcomes through the perceived superiority of AI over traditional methods. SEM suggests its indirect

impact, whereas SVM and Improved SVM demonstrate its significant direct influence on performance.

III. **Ease and Enjoyment of Use (EEU):** A key determinant of user engagement and satisfaction. SEM identifies EEU as a positive mediator, while SVM and Improved SVM reveal its strong predictive accuracy, mainly when AI tools are user-friendly and enjoyable.

IV. **AI Readiness and Facilitating Conditions (ARFC):** Essential for successful AI integration. SEM shows its moderate impact, with SVM and Improved SVM emphasizing the importance of institutional support and readiness in achieving positive academic outcomes.

V. **AI-induced Learning Anxiety (AILA):** A significant barrier to effective AI adoption is highlighted by SEM and SVM, both of which emphasize the negative impact on performance. The Improved SVM offers more reliable predictions by addressing multicollinearity.

VI. **Interactive Capability (IC):** Vital for enhancing engagement and interaction in ODL settings. SEM shows its significant role as a mediator, while SVM and Improved SVM confirm its strong influence on academic success.

VII. **Knowledge Absorption and User Satisfaction (KAUS):** This is the most significant predictor of academic performance. All models agree on its central role, with Improved SVM providing the most accurate predictions due to reduced multicollinearity.

VIII. **Systems Quality and Social Influence (SQSI):** Important for technical reliability and social support. SEM shows its moderate impact, while SVM and Improved SVM highlight its critical role in environments with high system quality and positive social influences.

The analysis of AI adoption factors using SEM, SVM, and Improved SVM reveals that these factors play varying but significant roles in influencing academic performance in ODL settings. SEM provides insights into the structural relationships and mediating effects among these factors. At the same time, SVM and Improved SVM offer robust predictive capabilities, with the latter addressing issues of multicollinearity to improve prediction accuracy and model stability. This comprehensive approach underscores the importance of a balanced and integrated strategy for AI adoption in education, where both the understanding of underlying relationships (as captured by SEM) and the focus on predictive accuracy (as highlighted by SVM and Improved SVM) are essential. The findings suggest that to maximize the positive impact of AI on student outcomes, educational institutions should focus on aligning AI tools with institutional goals, ensuring ease of use, providing robust support systems, and fostering positive social influences.

Future research should continue to explore these factors in more diverse educational contexts and investigate additional variables that may influence AI adoption and its effects on learning outcomes.

By doing so, the educational sector can better leverage AI technologies to enhance learning experiences and academic success in ODL settings.

4.5.6 Alignment of Research Findings with Research Questions, Hypotheses, and Objectives

The research findings of this study have been meticulously analysed to address the research questions, test the hypotheses, and achieve the specific objectives set out at the beginning of this thesis. The following discussion outlines how the results obtained align with these foundational elements of the research.

The first research question sought to identify the requirements for adopting AI in Open Distance Learning (ODL). The corresponding hypothesis posited that comprehensive identification of these requirements would enhance student academic performance. The results of this study, particularly through the analysis of AI readiness, ease of use, and knowledge absorption, have substantiated this hypothesis. The identification and fulfilment of these AI adoption requirements were shown to be crucial for improving student outcomes in ODL environments. This aligns with the first objective of designing a process framework that incorporates these factors to enhance the understanding of AI adoption in ODL. The second research question aimed to explore the design of a process model that effectively incorporates AI requirements into ODL. The corresponding hypothesis suggested that such a model would significantly enhance the understanding of AI adoption in ODL. The structural equation model (SEM) developed in this study has effectively captured the complex interactions between various AI adoption factors, providing a comprehensive framework that elucidates the dynamics of AI integration in ODL. This confirms the second hypothesis and achieves the objective of designing a research model that integrates AI adoption factors with student academic performance.

The third research question examined the design of a research model that incorporates AI factors and student academic performance. The hypothesis was that these AI adoption factors significantly impact student academic performance. The SEM analysis confirmed this hypothesis by demonstrating that factors such as knowledge absorption, user satisfaction, and interactive capability are strong predictors of student academic outcomes in ODL settings. This aligns with the third objective of developing a machine-learning model to predict the impact of these factors on student performance. The fourth research question addressed the development of machine learning models incorporating the impact factors of AI adoption and student academic performance. The corresponding hypothesis asserted that these models would effectively predict the impact of AI adoption on ODL students' academic performance. The Improved Support Vector Machine (SVM) developed in this study validated this hypothesis, as it demonstrated enhanced accuracy and stability in predictions,

particularly by addressing multicollinearity issues. This achievement aligns with the fourth objective of evaluating these models to establish their accuracy.

The fifth and final research question focused on evaluating the developed machine learning models to determine their level of accuracy. The hypothesis stated that such evaluations would have a significant impact on model accuracy. The results confirmed this hypothesis, showing that the Improved SVM, with its refined approach to multicollinearity, significantly improved the predictive performance of the models. This evaluation process has thus fulfilled the objective of enhancing the accuracy of the machine learning models used in the study. The research findings have thoroughly addressed the research questions, confirmed the hypotheses, and achieved the objectives set out at the beginning of this study. The developed frameworks and models provide a deeper understanding of AI adoption in ODL and offer robust tools for predicting and improving student academic performance. This comprehensive alignment underscores the success and significance of the research conducted in this thesis.

4.6 Implications of the results

The findings from the comparative analysis of Structural Equation Modelling (SEM), Support Vector Machine (SVM), and Improved SVM have profound implications for the adoption and application of Artificial Intelligence (AI) in Open and Distance Learning (ODL) settings. This section delves into the broader implications of these results, discussing their relevance for educational institutions, policymakers, educators, and future research. The implications are categorized into theoretical, practical, and policy-related impacts, each offering insights into how AI can be effectively leveraged to enhance academic performance in diverse educational contexts.

4.6.1 Theoretical Implications

The theoretical implications are as follows:

I. **Integration of Predictive and Structural Approaches:** The study's use of both SEM and SVM provides a comprehensive approach to understanding and predicting academic performance influenced by AI adoption factors. SEM's strength lies in its ability to validate theoretical models by establishing relationships between latent and observed variables, offering insights into the causal pathways that affect educational outcomes. SVM, on the other hand, excels in predictive accuracy, particularly in handling complex, non-linear relationships that SEM may not fully capture. By enhancing its capacity to address multicollinearity, the Improved SVM bridges the gap between these two approaches, ensuring both robust prediction and structural understanding. This integration underscores the importance of using

a multi-method approach in educational research, where theoretical validation and predictive modelling work hand-in-hand to provide a more holistic understanding of educational phenomena.

II. **Contribution to AI Adoption Theory:** The research contributes to the theoretical understanding of AI adoption in education by identifying key factors—such as AI Alignment and Relevance (AAR), Comparative Advantage of AI (CAAI), and Knowledge Absorption and User Satisfaction (KAUS)—that significantly influence academic performance. The findings validate the Technology Acceptance Model (TAM) and Unified Theory of Acceptance and Use of Technology (UTAUT) in the context of ODL, extending these models to incorporate AI-specific variables. This expansion provides a more nuanced theoretical framework for examining how AI technologies impact learning outcomes, particularly in distance education environments where traditional in-person interactions are limited.

III. **Understanding the Role of Moderators:** The study highlights the importance of considering moderating factors like gender and geographical location in AI adoption research. The Improved SVM results, in particular, show that these moderators can significantly influence the effectiveness of AI tools, with varying impacts on different demographic groups and regions. This insight contributes to the broader literature on educational equity and access, suggesting that AI adoption strategies should be tailored to address the specific needs and challenges of diverse student populations.

4.6.2 Practical Implications

The practical implications are as follows:

I. **Enhancing AI Integration in ODL Settings:** The results suggest several practical steps that educational institutions can take to improve the integration of AI in ODL settings. First, institutions should focus on aligning AI tools with both institutional goals and student needs (as highlighted by the importance of AAR in the models). This alignment ensures that AI technologies are not only adopted but also effectively utilized to enhance learning outcomes. Second, improving the ease of use and enjoyment of AI tools (EEU) can significantly boost student engagement, leading to better academic performance. Institutions should, therefore, invest in user-friendly AI interfaces and provide adequate training to help students overcome any learning anxiety (AILA) associated with new technologies.

II. **Addressing Multicollinearity in Educational Data:** The findings underscore the importance of addressing multicollinearity in educational data analysis, mainly when using predictive

models like SEM and SVM. The Improved SVM model, which incorporates VIF optimization, demonstrates that reducing multicollinearity leads to more stable and reliable predictions. Educational practitioners and researchers should, therefore, consider incorporating techniques to manage multicollinearity when developing predictive models for student performance, ensuring that the resulting insights are both accurate and actionable.

III. **Tailoring AI Solutions to Diverse Student Populations:** The study's findings on the moderating effects of gender and geographical location suggest that AI solutions should be customized to meet the needs of diverse student groups. For instance, the improved performance of AI tools for female students in specific contexts implies that gender-specific support and content may enhance learning outcomes. Similarly, the differential impact of AI tools in various geographical locations suggests that local contexts—such as access to technology and cultural attitudes towards AI—should be considered when implementing AI in education. By tailoring AI solutions to these specific needs, educational institutions can maximize the effectiveness of their AI initiatives.

IV. **Prioritizing Quality and Social Influence in AI Adoption:** The significant role of Systems Quality and Social Influence (SQSI) in the study implies that the success of AI adoption in ODL is heavily dependent on the technical reliability of the AI systems and the social environment in which they are used. Educational institutions should prioritize deploying high-quality AI systems that are reliable and efficient, ensuring that these tools meet the technical standards necessary for effective educational delivery. Additionally, fostering a positive social environment where peers and instructors support AI adoption can significantly enhance the overall effectiveness of AI in improving academic performance.

V. **Leveraging AI for Early Intervention and Support:** Predicting academic performance through AI enables institutions to identify at-risk students early, providing an opportunity for timely intervention. By accurately predicting which students might struggle, institutions can tailor support services to these students, offering targeted resources such as tutoring, mentoring, or additional academic assistance. This early intervention can significantly improve the learning experience, potentially reducing dropout rates and enhancing overall program completion rates. As AI tools continue to evolve, their ability to provide predictive insights will become increasingly valuable in supporting student success and ensuring that all students have the opportunity to achieve their academic goals.

4.6.3 Policy Implications

The policy implications are as follows:

I. **Supporting AI Readiness and Facilitating Conditions:** Policymakers should focus on

creating a supportive environment for AI adoption in education by ensuring that both students and institutions are adequately prepared. This includes investing in infrastructure that supports AI technologies, providing funding for training programs to improve AI readiness (ARFC), and developing policies that facilitate the widespread adoption of AI tools in educational settings. By enhancing AI readiness, policymakers can help reduce barriers to AI adoption, leading to more equitable and effective educational outcomes.

II. **Promoting Equity in AI Adoption:** The study's findings on the moderating effects of gender and geographical location highlight the need for policies that address disparities in AI adoption and usage. Policymakers should ensure that AI technologies are accessible to all students, regardless of gender, geographical location, or socioeconomic background. This may involve targeted interventions, such as providing additional resources and support to underrepresented groups or regions, to ensure that the benefits of AI are equitably distributed.

III. **Encouraging Evidence-Based AI Integration:** Policymakers should advocate for the use of evidence-based practices in the integration of AI into education. The study's comparative analysis of SEM, SVM and Improved SVM provides a strong case for the importance of using rigorous analytical methods to assess the impact of AI on educational outcomes. Policies that encourage the adoption of such methods can help ensure that AI tools are implemented in ways that are both effective and scientifically validated, leading to better educational outcomes at scale.

IV. **Fostering Collaboration Between Stakeholders:** The successful adoption of AI in education requires collaboration between multiple stakeholders, including educational institutions, technology providers, policymakers, and researchers. The findings suggest that such collaboration is essential for addressing the complex challenges associated with AI adoption, from managing multicollinearity in predictive models to ensuring that AI tools are aligned with educational goals. Policymakers should, therefore, promote partnerships between these stakeholders to facilitate the development and implementation of AI solutions that are both innovative and effective.

4.6.4 Implications for Future Research

The implications for future research are as follows:

I. **Expanding the Scope of AI Adoption Studies:** The study's findings open several avenues for future research. One important direction is to expand the scope of AI adoption studies to include a broader range of educational contexts and demographic groups. This includes

examining how AI adoption factors influence academic performance in different types of educational institutions (e.g., primary vs. tertiary education) and among various student populations (e.g., adult learners and students with disabilities).

II. **Investigating Longitudinal Effects:** Future research should also investigate the long-term effects of AI adoption on academic performance. While this study provides valuable insights into the immediate impact of AI adoption factors, understanding how these effects evolve would provide a more comprehensive picture of AI's role in education. Longitudinal studies could explore how sustained use of AI tools influences learning outcomes, student satisfaction, and educational equity.

III. **Exploring New AI Adoption Factors:** The study identifies several key AI adoption factors, but future research could explore additional variables that may influence AI's impact on education. For example, factors related to AI ethics, data privacy, and student autonomy could be critical in understanding the broader implications of AI adoption. By expanding the range of variables studied, researchers can develop a more nuanced understanding of the conditions under which AI is most effective in educational settings.

IV. **Combining Quantitative and Qualitative Approaches:** Finally, future research should consider combining quantitative methods, like SEM and SVM, with qualitative approaches to provide a richer understanding of AI adoption in education. While quantitative models offer valuable predictive and structural insights, qualitative research can capture the lived experiences of students and educators, providing context and depth to the findings. This mixed-methods approach could lead to more holistic and actionable recommendations for AI adoption in education.

The implications of this study are far-reaching, offering valuable insights for theory, practice, policy, and future research in the field of AI adoption in education. By understanding the factors that influence AI's impact on academic performance, educational institutions, policymakers, and researchers can develop strategies that maximize the benefits of AI in ODL settings. The study underscores the importance of a comprehensive, evidence-based approach to AI integration, one that considers the unique needs of diverse student populations and the complex dynamics of educational environments.

4.7 Benchmark of the Results

Benchmarking the results from this study involves comparing the outcomes obtained from the analysis of AI adoption factors using Structural Equation Modelling (SEM), Support Vector Machine (SVM), and Improved SVM against established standards, previous studies, and industry expectations. The purpose of this benchmarking is to assess the reliability, accuracy, and generalizability of the findings, as well as to highlight the contributions of this research in the context of existing literature. This section provides an extensive analysis of how the results align with or differ from prior research, how they measure up against industry benchmarks, and the implications of these comparisons for future research and practical applications in Open and Distance Learning (ODL) settings.

4.7.1 Benchmarking Against Established Theoretical Models

I. **Validation of AI Adoption Factors:** The AI adoption factors identified in this study—such as AI Alignment and Relevance (AAR), Comparative Advantage of AI (CAAI), and Ease and Enjoyment of Use (EEU)—have been benchmarked against established theoretical models like the Technology Acceptance Model (TAM) and the Unified Theory of Acceptance and Use of Technology (UTAUT). These models have long been used to understand technology adoption in various contexts, including education.

- **TAM and UTAUT Comparison:** The results from SEM validate the theoretical constructs of TAM and UTAUT, particularly in how AAR and CAAI contribute to perceived usefulness and ease of use, which are core components of TAM. This alignment suggests that the AI adoption factors identified in this study are consistent with established theories, thereby reinforcing their relevance and applicability in ODL settings (Strzelecki, 2023; Dwivedi et al., 2017). The benchmarking against these models shows that while AI-specific factors are critical, they do not diverge significantly from broader technology acceptance theories but rather expand on them to suit the nuances of AI in education.

II. **Enhanced Predictive Modelling with SVM:** The use of SVM and Improved SVM in this study provides a benchmark for predictive modelling in educational research. Traditional models like SEM are well-suited for understanding relationships between variables, but SVM offers enhanced predictive accuracy, particularly in handling non-linear relationships and complex interactions.

- **Predictive Accuracy Benchmarking:** When compared to traditional statistical models used in educational research, SVM demonstrates superior performance in predicting academic outcomes based on AI adoption factors. By focusing on reducing its multicollinearity, the Improved SVM sets a new benchmark for predictive modelling by

enhancing the stability and reliability of predictions (Zhang, 2021). This marks a significant advancement over previous studies that relied solely on SEM or other regression-based models, showcasing the value of integrating machine learning approaches in educational research.

4.7.2 Comparison with Previous Studies

I. **Alignment with Prior Research on AI in Education:** The findings from this study align with previous research that highlights the importance of AI adoption in improving educational outcomes. Studies by scholars such as Nguyen (2023) and Holmes et al. (2023) have emphasized the potential of AI to transform educational practices, particularly in ODL environments.

- **Benchmarking Educational Impact:** The results from this study corroborate the positive impact of AI adoption factors like KAUS and EEU on student performance, which have been similarly highlighted in prior research (Nguyen, 2023; Holmes et al., 2023). However, this study goes further by providing a detailed analysis of how these factors interact with moderating variables such as gender and geographical location, offering a more nuanced understanding of AI's impact that previous studies have not fully explored. This positions the current research as a benchmark for future studies that seek to explore the complexities of AI adoption in diverse educational contexts.

II. **Divergence from Traditional Educational Research:** While the results align with some aspects of prior research, they also diverge in significant ways, particularly in the emphasis on predictive modelling and the management of multicollinearity.

- **Handling of Multicollinearity:** Previous studies in educational research often struggled with issues of multicollinearity, leading to less reliable models and predictions. The Improved SVM's approach to VIF optimization sets a new standard for addressing this issue, ensuring that the predictive models used in this study are both accurate and robust (Chan et al., 2022). This divergence from traditional methods highlights the innovative contributions of this research, particularly in advancing the use of machine-learning techniques in educational settings.

4.7.3 Industry Benchmarking

I. **Alignment with Industry Standards for AI Implementation:** The study's findings can be benchmarked against industry standards for AI implementation in education, as set by organizations like UNESCO and the International Society for Technology in Education

(ISTE). These standards emphasize the need for AI tools to be aligned with educational goals, user-friendly, and equitable.

- **Benchmarking Against UNESCO and ISTE Standards:** The emphasis on AAR and EEU in this study aligns well with UNESCO's guidelines for AI in education, which stress the importance of aligning AI tools with institutional goals and ensuring they are accessible and user-friendly (UNESCO, 2022). The study's findings regarding the importance of system quality and social influence (SQSI) further support the ISTE standards, which advocate for high-quality, reliable AI systems that enhance learning environments (ISTE, 2021). The alignment with these standards indicates that the results of this study not only contribute to academic research but also have practical implications for industry practices and policies related to AI in education.

II. **Benchmarking Predictive Models Against Industry Expectations:** In the tech industry, particularly in fields like educational technology, the accuracy and reliability of predictive models are critical benchmarks for success. The use of SVM and Improved SVM in this study provides a benchmark for how AI adoption factors can be modelled to predict academic outcomes.

- **Predictive Model Performance:** The predictive performance of SVM and Improved SVM in this study can be benchmarked against industry expectations for machine learning models in educational settings. The relatively low error rates (MAE, MSE, RMSE) observed in the SVM models indicate that these approaches meet or exceed industry standards for predictive accuracy, positioning them as viable tools for real-world educational applications (Ojajuni et al., 2021; Leeuwenberg et al., 2022). This sets a new benchmark for how educational institutions and technology providers can use predictive modelling to enhance AI adoption strategies and improve student outcomes.

4.7.4 Benchmarking Within the Context of Open and Distance Learning (ODL)

I. **Addressing Challenges in ODL:** The study's focus on ODL settings provides a benchmark for how AI can address specific challenges associated with remote education, such as student engagement, access to resources, and the quality of interactions.

- **ODL-Specific Benchmarks:** The study's findings that factors like interactive capability (IC) and AI readiness (ARFC) significantly influence academic performance in ODL environments set a benchmark for future research and practice in this area. These results suggest that for AI to be effective in ODL, it must be not only technically robust but also capable of enhancing the quality of interactions and providing adequate support for both

students and educators (Akinwalere & Ivanov, 2022; Rakya, 2023). This benchmark emphasizes the need for AI tools that are specifically designed or adapted for the unique challenges of ODL.

II. **Enhancing Student Outcomes in ODL:** One of the critical benchmarks for AI in education is its ability to improve student outcomes. This study's analysis of AI adoption factors against academic performance provides a benchmark for evaluating the effectiveness of AI interventions in ODL settings.

- **Student Performance Benchmarking:** The study demonstrates that AI tools that align with student needs, offer comparative advantages over traditional methods, and are easy to use can significantly enhance academic performance in ODL environments (Xu, 2024). This finding provides a benchmark for educational institutions to measure the success of their AI implementations, guiding them in selecting and deploying AI tools that are most likely to improve student outcomes in remote learning contexts.

Table 4.11 organizes the benchmark results for AI's role in ODL, for easier reference and analysis.

Table 4.11. Benchmarking AI in Open and Distance Learning (ODL) Based on Study Findings

Benchmark Category	Study Findings	References
Addressing Challenges in ODL	AI can address ODL-specific challenges like student engagement, access to resources, and interaction quality.	Akinwalere & Ivanov (2022), Rakya (2023)
ODL-Specific Benchmarks	Factors like Interactive Capability (IC) and AI Readiness (ARFC) significantly influence academic performance in ODL.	Akinwalere & Ivanov (2022), Rakya (2023)
Enhancing Student Outcomes in ODL	AI tools that align with student needs, offer comparative advantages, and are user-friendly can significantly improve academic performance.	Xu (2024)
Student Performance Benchmarking	AI interventions that are technically robust and capable of enhancing interactions set a benchmark for future implementations in ODL.	Akinwalere & Ivanov (2022), Xu (2024), Rakya (2023)

4.7.5 Implications of Benchmarking for Future Research

I. **Establishing New Standards for AI Research:** The benchmarking of results from this study against theoretical models, previous research, industry standards, and ODL-specific challenges establishes new standards for AI research in education. Future studies can build on these benchmarks to explore new AI adoption factors, refine predictive models, and further investigate the role of moderating variables in different educational contexts.

- **Setting Research Agendas:** The benchmarks established in this study can guide future research agendas, particularly in areas such as the integration of machine learning techniques with traditional educational models, the exploration of AI's impact across diverse student populations, and the development of AI tools tailored to specific

educational environments like ODL (Abbas et al., 2023).

II. **Encouraging the Adoption of Best Practices:** By providing clear benchmarks for AI adoption in education, this study encourages the adoption of best practices in both research and practice. Educational institutions can use these benchmarks to evaluate their AI initiatives, ensuring that they align with the most effective strategies identified in the research.

- **Best Practices in AI Adoption:** The benchmarks related to the importance of aligning AI tools with institutional goals, enhancing system quality, and addressing multicollinearity in predictive models can serve as best practices for both researchers and practitioners (Strzelecki, 2023; Leeuwenberg et al., 2022). By adopting these practices, educational institutions can maximize the benefits of AI, leading to improved student outcomes and more effective educational processes.

The benchmarking of results in this study provides a comprehensive assessment of how the findings compare to existing theoretical models, previous research, industry standards, and the specific needs of ODL settings. By establishing new benchmarks in these areas, the study not only contributes to the academic literature but also offers practical guidelines for the effective adoption and implementation of AI in education. These benchmarks serve as valuable references for future research and practice, helping to shape the development of AI tools that are both innovative and impactful in enhancing educational outcomes.

This benchmarking analysis highlights the importance of a multi-method approach in AI research, especially in education. The combination of SEM, SVM, and Improved SVM offers both theoretical depth and strong predictive capabilities, setting a new standard for future studies. It emphasizes the need for AI solutions tailored to the unique challenges of ODL, providing a roadmap for educators, policymakers, and technology providers in AI adoption. As AI's role in education grows, these benchmarks will guide policy, practice, and research, ensuring AI tools are practical, equitable, accessible, and aligned with 21st-century educational goals. By refining these benchmarks, the educational community can fully leverage AI to enhance learning outcomes for all students, regardless of location or background.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary

This study aimed to develop a robust process framework for predicting the impact of Artificial Intelligence (AI) adoption on students' academic performance in Open and Distance Learning (ODL) environments. The research focused on several key objectives. To achieve this, we designed a process framework, developed predictive models using a Support Vector Machine (SVM), and evaluated these models to determine their accuracy in predicting academic outcomes.

The study successfully developed a comprehensive process framework that integrates AI adoption factors, considers moderating variables such as gender and geographical differences, and applies machine learning techniques to predict academic performance. The framework underwent rigorous testing through multiple stages, including data pre-processing, model training, and validation, which ensured the reliability of the predictive models. The findings confirmed that AI adoption positively influences academic performance when factors such as ease of use, knowledge absorption, and user satisfaction are adequately addressed. Moreover, the study highlighted the importance of customizing AI tools to cater to the diverse needs of ODL students across different regions.

5.2 Conclusion

The exponential progression in artificial intelligence (AI) and its integration into Education has generated significant interest among researchers. One crucial aspect is the impact of AI adoption on students' academic performance, particularly in the context of Online Distance Learning (ODL). This study has developed a process framework utilising a Support Vector Machine (SVM) to predict the impact of AI adoption on students' academic performance in ODL. The theoretical framework is the cornerstone of any research, laying the foundation for interpreting the dynamics and outcomes of the study. In the context of this present work, the theoretical framework is instrumental in guiding the exploration and analysis of critical components such as AI adoption factors, moderating factors, and the outcome variable of students' academic performance.

5.3 Recommendations

The research concludes that the adoption of AI in ODL can significantly enhance students' academic performance when adequately aligned with institutional goals and tailored to meet the specific needs of students. The developed process framework and predictive models provide valuable insights into how AI adoption factors, coupled with moderating variables, influence academic outcomes. By leveraging SVM, the study has demonstrated the effectiveness of machine learning in forecasting

student performance, offering a reliable tool for educators and policymakers to optimize AI integration in educational settings. The successful development and validation of the predictive framework underscore the importance of considering both technical and non-technical factors in AI adoption. The findings suggest that a one-size-fits-all approach is insufficient; instead, AI tools should be adapted to the unique challenges and opportunities within ODL systems. This study contributes to the broader discourse on AI in education by providing a structured approach to understanding and predicting the impact of AI on learning outcomes.

5.4 Contributions to Knowledge

The contributions to the knowledge of this study are as follows:

I. **In-Depth Requirements Elicitation Report**: This study produces a comprehensive report that meticulously identifies and analyzes the key factors influencing Artificial Intelligence (AI) adoption in the context of students' academic performance in Open Distance Learning (ODL). This report delves into various dimensions, such as technological, pedagogical, and institutional factors that contribute to the effective integration of AI in educational settings. Doing so provides a foundational understanding of the current landscape of AI adoption in the educational sector, highlighting both the opportunities and challenges.

II. **Robust Process Framework for AI Adoption in ODL**: The study proposes a novel process framework specifically designed for AI adoption in ODL environments based on the insights gained from the requirements elicitation. This framework outlines a structured and strategic approach, incorporating the identified factors to facilitate a more effective and seamless adoption of AI technologies in ODL. It serves as a guide for educational institutions aiming to leverage AI to enhance teaching and learning experiences.

III. **Comprehensive Research Model on AI Adoption and Student Performance**: A pivotal contribution of this study is creating an encompassing research model that integrates the critical factors influencing AI adoption with their subsequent impact on student academic performance in ODL. This model aims to fill existing gaps in the literature by providing a holistic view of how AI technologies can influence educational outcomes. It is an essential resource for future research and practice in educational technology.

IV. **Advanced Machine Learning Models for Predicting Academic Performance**: The study also focuses on developing cutting-edge machine learning models that utilize the factors identified from AI adoption to predict their effects on student academic performance in ODL. These models are designed to process complex datasets and provide predictive insights, serving as invaluable tools for educational administrators and policymakers to make data-driven decisions.

V. **Detailed Model Evaluation Report**: An integral part of this study is thoroughly evaluating the developed machine learning models. This report assesses the models' accuracy, reliability, and applicability in real-world educational settings. It critically analyses the models' strengths and limitations, offering recommendations for improvement and future development. This evaluation is crucial for education sector stakeholders considering the practical deployment of AI-based predictive models.

VI. **Improved Support Vector Machine (SVM) with VIF Optimization**: This study contributes by developing and refining an Improved Support Vector Machine (SVM) model that incorporates Variance Inflation Factor (VIF) optimization, utilizing internal consistency and reliability checks through Cronbach's Alpha. This enhancement is designed to improve the model's stability, reliability, and ability to discern the individual impact of AI adoption factors on students' academic performance in ODL. By addressing multicollinearity issues, the VIF optimization ensures a more robust and stable model. This contribution is particularly significant for educational stakeholders, providing an advanced tool for more precise and dependable predictions of student outcomes based on critical AI adoption factors.

Through these contributions, the study aims to significantly advance the understanding of AI adoption in ODL and its impact on student academic performance. It offers practical tools and models for educators and policymakers and sets the stage for future innovations in the field.

5.5 Future Research Directions

The findings from this study provide a solid foundation for future research in the area of Artificial Intelligence (AI) adoption in Open and Distance Learning (ODL) environments. However, there remain several avenues for further exploration that could enhance our understanding of AI's impact on education and refine the predictive frameworks developed in this research.

I. **Exploration of Additional AI Adoption Factors:** Future research should investigate other factors that may influence the successful adoption of AI in ODL. For instance, cultural attitudes towards technology, the role of instructor training and preparedness, and the impact of social learning networks could be significant. Understanding these additional factors can help create more comprehensive models that capture the full range of variables affecting AI integration in education.

II. **Longitudinal Studies on AI's Impact:** While this study provides a snapshot of AI's influence on academic performance, longitudinal research is needed to understand the long-term effects of AI adoption in ODL. Future studies should track cohorts of students over extended periods to assess how sustained interaction with AI tools affects learning outcomes, retention rates,

and overall academic success. Such research could also explore how students' perceptions and usage of AI evolve.

III. **Integration of Emerging Technologies:** As technology continues to evolve, future research should explore the integration of other emerging technologies alongside AI in ODL settings. For example, the potential of Virtual Reality (VR), Augmented Reality (AR), and Blockchain to enhance educational experiences and improve academic outcomes should be examined. These technologies could complement AI by providing more immersive and secure learning environments.

IV. **Customization of AI Tools for Diverse Learning Needs:** Future research should focus on how AI tools can be further customized to meet the needs of diverse learner groups, including students with disabilities, non-traditional learners, and those in underserved regions, building on the findings of this study. Research should explore the development of adaptive AI systems that can personalize learning experiences based on individual student profiles, learning styles, and progress.

V. **Cross-Cultural Comparisons of AI Adoption:** Future studies should conduct cross-cultural comparisons of AI adoption in ODL environments to gain a more global perspective. By examining how different educational systems and cultural contexts influence AI's impact, researchers can identify best practices and potential challenges that are unique to specific regions or demographics. This could lead to more tailored approaches to AI integration across diverse educational settings.

VI. **Ethical Considerations and AI in Education:** As AI continues to play a larger role in education, it is crucial to address the ethical implications of its use. Future research should explore issues related to data privacy, algorithmic bias, and the potential for AI to exacerbate existing inequalities in education. Developing ethical guidelines and frameworks for AI in education will be essential to ensure that its adoption benefits all students equitably.

VII. **Enhancement of Predictive Models:** The predictive models developed in this research, while robust, could be further refined to improve their accuracy and generalizability. Future research should explore the integration of more advanced machine learning algorithms, such as deep learning, and the inclusion of additional data sources, such as real-time learning analytics, to enhance the predictive power of these models. Additionally, researchers could explore the application of these models in different educational contexts, such as vocational training or professional development programs.

Addressing these future research directions will help advance AI in education, ensuring its ethical and practical use to enhance learning outcomes and provide equitable opportunities for all students.

References

Abbas, N., Ali, I., Manzoor, R., Hussain, T., & Hussain, M. H. a. I. (2023). Role of artificial intelligence tools in enhancing students' educational performance at higher levels. *Journal of Artificial Intelligence Machine Learning and Neural Network*, *35*, 36–49. https://doi.org/10.55529/jaimlnn.35.36.49.

Adewale, M.D. et al. (2024). Comparative Performance Evaluation of Random Forest, Extreme Gradient Boosting and Linear Regression Algorithms Using Nigeria's Gross Domestic Products. In: Seeam, A., Ramsurrun, V., Juddoo, S., Phokeer, A. (eds) Innovations and Interdisciplinary Solutions for Underserved Areas. InterSol 2023. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 541. Springer, Cham. https://doi.org/10.1007/978-3-031-51849-2_9

Aftarczuk, K. (2007). "Evaluation of selected data mining algorithms implemented in Medical Decision Support Systems," Blekinge Institute of Technology School of Engineering, Blekinge.

Akinwalere, S. N., & Ivanov, V. (2022). Artificial intelligence in higher Education: challenges and opportunities. *BORDER CROSSING*, *12*(1), 1–15. https://doi.org/10.33182/bc.v12i1.2015.

Akyuz, Y. (2020). Effects of Intelligent Tutoring Systems (ITS) on Personalized Learning (PL). *Creative Education*, *11*(06), 953–978. https://doi.org/10.4236/ce.2020.116069

Alam, S. S., Masukujjaman, M., Ahmad, M., & Jaffor, R. (2022). Acceptance of online distance learning (ODL) among students: Mediating role of utilitarian and hedonic value. *Education and Information Technologies*. https://doi.org/10.1007/s10639-022-11533-3

Albers, C. J., & Lakens, D. (2018). When power analyses based on pilot data are biased: Inaccurate effect size estimators and follow-up bias. *Journal of Experimental Social Psychology*, *74*, 187–195. https://doi.org/10.1016/j.jesp.2017.09.004

Allam, S. N. S., Hassan, M. A., Mohideen, R. S., Ramlan, A. F., & Kamal, R. M. (2020). Online Distance Learning Readiness During Covid-19 Outbreak Among Undergraduate Students. *International Journal of Academic Research in Business & Social Sciences*, *10*(5). https://doi.org/10.6007/ijarbss/v10-i5/7236

Almaiah, M. A., Alfaisal, R., Salloum, S. A., Hajjej, F., Shishakly, R., Lutfi, A., Alrawad, M., Mulhem, A. A., Alkhdour, T., & Al-Maroof, R. S. (2022). Measuring Institutions' Adoption of Artificial Intelligence Applications in Online Learning Environments: Integrating the Innovation Diffusion Theory with Technology Adoption Rate. *Electronics*, *11*(20), 3291. https://doi.org/10.3390/electronics11203291

Ali, O., Abdelbaki, W., Shrestha, A., Elbasi, E., Alryalat, M. a. A., & Dwivedi, Y. K. (2023). A systematic literature review of artificial intelligence in the healthcare sector: Benefits, challenges, methodologies, and functionalities. *Journal of Innovation & Knowledge*, *8*(1), 100333. https://doi.org/10.1016/j.jik.2023.100333

Almaiah, M. A., Alamri, M. M., and Al-Rahmi, W. (2019). Applying the UTAUT model to explain the students' acceptance of mobile learning system in higher Education. IEEE Access 7, 174673–174686. doi: 10.1109/ACCESS.2019.29 57206

Almaiah, M. A., Alfaisal, R., Salloum, S. A., Hajjej, F., Shishakly, R., Lutfi, A., Alrawad, M., Mulhem, A. A., Alkhdour, T., & Al-Maroof, R. S. (2022). Measuring Institutions' Adoption of Artificial Intelligence Applications in Online Learning Environments: Integrating the Innovation Diffusion Theory with Technology Adoption Rate. *Electronics*, *11*(20), 3291. https://doi.org/10.3390/electronics11203291

Almaiah, M. A., Alfaisal, R., Salloum, S. A., Hajjej, F., Thabit, S., El-Qirem, F. A., . . . Al-Maroof, R. S. (2022). Examining the Impact of Artificial Intelligence and Social and Computer Anxiety in E-Learning Settings: Students' Perceptions at the University Level. Electronics, 11(22), 3662. https://doi.org/10.3390/electronics11223662

Alonso, M., Rubio, A. V., Escrig, T., Soto, T., Serrano-Lanzarote, B., & Matarredona-Desantes, N. (2021). Identification of Measures to Strengthen Resilience in Homes on the Basis of Lockdown Experience during COVID-19. *Sustainability*, *13*(11), 6168. https://doi.org/10.3390/su13116168

Alqahtani, M. (2021). Predicting Student Performance in Online Learning Using Machine Learning

Techniques. Journal of Educational Computing Research, 59(6), 1427-1449. https://doi.org/10.1177/0735633120968937
An, C., Lim, H., Kim, D., Chang, J. M., Choi, Y. Y., & Kim, S. (2020). Machine learning prediction for mortality of patients diagnosed with COVID-19: a nationwide Korean cohort study. *Scientific Reports*, *10*(1). https://doi.org/10.1038/s41598-020-75767-2
Ameri, A., Khajouei, R., Ameri, A., and Jahani, Y. (2020). Acceptance of a mobile-based educational application (labsafety) by pharmacy students: an application of the UTAUT2 model. Educ. Inf. Technol. 25, 419–435. doi: 10.1007/s10639-019-09965-5
Asif, R., Merceron, A., Ali, S. F., & Haider, N. G. (2017). Analyzing undergraduate students' performance using educational data mining. *Computers & Education*, *113*, 177–194. https://doi.org/10.1016/j.compedu.2017.05.007
Au, O., Li, K., & Wong, T. Y. (2018). Student persistence in open and distance learning: success factors and challenges. *AAOU Journal*, *13*(2), 191–202. https://doi.org/10.1108/aaouj-12-2018-0030
Ayouni, S., Hajjej, F., Maddeh, M., & Al-Otaibi, S. (2021). A new ML-based approach to enhance student engagement in online environment. *PLOS ONE*, *16*(11), e0258788. https://doi.org/10.1371/journal.pone.0258788
Babić, I. (2017). Machine learning methods in predicting the student academic motivation. *Croatian Operational Research Review*, *8*(2), 443–461. https://doi.org/10.17535/crorr.2017.0028
Bajaj, A. (2023). Performance Metrics in Machine Learning [Complete Guide]. *neptune.ai*. https://neptune.ai/blog/performance-metrics-in-machine-learning-complete-guide
Bernacki, M. L., Chavez, M. M., & Uesbeck, P. M. (2020). Predicting achievement and providing support before STEM majors begin to fail. *Computers & Education*, *158*, 103999. https://doi.org/10.1016/j.compedu.2020.103999
Bertl, M., Metsallik, J., & Ross, P. (2022). A systematic literature review of AI-based digital decision support systems for post-traumatic stress disorder. *Frontiers in Psychiatry*, *13*. https://doi.org/10.3389/fpsyt.2022.923613
Birajdar, D., & Vasudevan, H. (2022). Critical Success Factors for Industry 4.0 Readiness and Adoption: A Conceptual Framework for Indian Manufacturing Industries. In *IOS Press eBooks*. https://doi.org/10.3233/atde220743
Bozkurt, A., Karadeniz, A., Baneres, D., Rodríguez, M. E., & Rodríguez, M. E. (2021, January 15). *Artificial Intelligence and Reflections from Educational Landscape: A Review of AI Studies in Half a Century*. Sustainability. https://doi.org/10.3390/su13020800
Buenaño-Fernández, D., Gil, D., & Luján-Mora, S. (2019). Application of Machine Learning in Predicting Performance for Computer Engineering students: A case study. *Sustainability*, *11*(10), 2833. https://doi.org/10.3390/su11102833
Chan, J. Y., Leow, S. M. H., Bea, K. T., Cheng, W. K., Phoong, S. W., Hong, Z., & Chen, Y. (2022). Mitigating the multicollinearity Problem and its machine Learning Approach: A review. *Mathematics*, *10*(8), 1283. https://doi.org/10.3390/math10081283.
Charness, N., & Boot, W. R. (2016). Technology, Gaming, and Social Networking. Handbook of the Psychology of Aging, 389–407. https://doi.org/10.1016/b978-0-12-411469-2.00020-0
Chaudhry, M. A., & Kazim, E. (2021). Artificial Intelligence in Education (AIEd): a high-level academic and industry note 2021. AI And Ethics, 2(1), 157–165. https://doi.org/10.1007/s43681-021-00074-z
Chaudhary, S. C., & Dey, N. (2013). Assessment in Open and Distance Learning System (ODL): A Challenge. *Open Praxis*, *5*(3), 207. https://doi.org/10.5944/openpraxis.5.3.65
Chen, L., Chen, P., & Lin, Z. (2020). Artificial Intelligence in Education: A Review. IEEE Access, 8, 75264–75278. https://doi.org/10.1109/access.2020.2988510
Chen, X., Xie, H., & Hwang, G. (2020). A multi-perspective study on Artificial Intelligence in Education: grants, conferences, journals, software tools, institutions, and researchers. Computers & Education: Artificial Intelligence, 1, 100005. https://doi.org/10.1016/j.caeai.2020.100005
Chen, X., Xie, H., Zou, D., & Hwang, G. J. (2020). Application and theory gaps during the rise of

Artificial Intelligence in Education. *Computers and Education: Artificial Intelligence*, *1,* 100002. doi:10.1016/j.caeai.2020.100002

Chopra, D., & Khurana, R. (2023). Support Vector Machine. In *BENTHAM SCIENCE PUBLISHERS eBooks* (pp. 58–73). https://doi.org/10.2174/9789815124422123010006

Cruz-Jesus, F., Castelli, M., Oliveira, T., Mendes, R. E., Nunes, C. S., Sa-Velho, M., & Rosa-Louro, A. (2020). Using artificial intelligence methods to assess academic achievement in public high schools of a European Union country. *Heliyon*, *6*(6), e04081. https://doi.org/10.1016/j.heliyon.2020.e04081

Dabingaya, M. (2022). Analyzing the Effectiveness of AI-Powered Adaptive Learning Platforms in Mathematics Education. *Interdisciplinary Journal Papier Human Review*, *3*(1), 1–7. https://doi.org/10.47667/ijphr.v3i1.226

Daraz, L., Bouseh, S., & Chang, B. S. (2022). Subpar: The Challenges of Gender Parity in Canada's Artificial Intelligence Ecosystem. *Computer and Information Science*, *15*(2), 1. https://doi.org/10.5539/cis.v15n2p1

de la Torre-López, J., Ramírez, A. & Romero, J.R. (2023). Artificial intelligence to automate the systematic review of scientific literature. Computing https://doi.org/10.1007/s00607-023-01181-x

Demir, Ö., & Yurdugül, H. (2015). The Exploration of Models Regarding E-Learning Readiness: Reference Model Suggestions. International Journal of Progressive Education, v11 n1 p173-194. https://eric.ed.gov/?id=EJ1060907

Dwivedi, Y. K., Rana, N. P., Jeyaraj, A., Clement, M., & Williams, M. D. (2017). Re-examining the Unified Theory of Acceptance and Use of Technology (UTAUT): towards a revised theoretical model. Information Systems Frontiers, 21(3), 719–734. https://doi.org/10.1007/s10796-017-9774-y.

Dua, A. (2021). Applications of artificial intelligence in open and distance learning. *Techno Learn : An International Journal of Educational Technology*, *11*(2). https://doi.org/10.30954/2231-4105.02.2021.1

European Commission (2018), Commission communication – Artificial intelligence for Europe, Com, 237 final.

Faul, F., Erdfelder, E., Buchner, A., & Lang, A. (2009). Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses. *Behavior Research Methods*, *41*(4), 1149–1160. https://doi.org/10.3758/brm.41.4.1149

Fernández, A., García, S., & Herrera, F. (2011). Addressing the Classification with Imbalanced Data: Open Problems and New Challenges on Class Distribution. In *Lecture notes in computer science* (pp. 1–10). https://doi.org/10.1007/978-3-642-21219-2_1

Gao, H. (2022). Online AI-Guided Video Extraction for Distance Education with Applications. *Mathematical Problems in Engineering*, *2022*, 1–7. https://doi.org/10.1155/2022/5028726

García-Martínez, I., Batanero, J. M. F., Fernández-Cerero, J., & León, S. P. (2023). Analysing the Impact of artificial intelligence and computational sciences on student performance: systematic review and meta-analysis. *Journal of New Approaches in Educational Research*, *12*(1), 171. https://doi.org/10.7821/naer.2023.1.1240

Gardner, J., Brooks, C., & Baker, R. S. (2019). *Evaluating the Fairness of Predictive Student Models Through Slicing Analysis*. https://doi.org/10.1145/3303772.3303791

Ghojogh, B., & Crowley, M. (2019). The Theory Behind Overfitting, Cross Validation, Regularization, Bagging, and Boosting: Tutorial. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1905.12787

Gligorea, I., Cioca, M., Oancea, R., Gorski, A., Gorski, H., & Tudorache, P. (2023). Adaptive Learning Using Artificial Intelligence in e-Learning: A Literature review. *Education Sciences*, *13*(12), 1216. https://doi.org/10.3390/educsci13121216

Goswami, A., & Dutta, S. (2016). Gender Differences in Technology Usage—A Literature Review. *Open Journal of Business and Management*, *04*(01), 51–59. https://doi.org/10.4236/ojbm.2016.41006

Haenlein, M., & Kaplan, A. (2019). A Brief History of artificial intelligence: on the past, present, and future of artificial intelligence. *California Management Review*, *61*(4), 5–14. https://doi.org/10.1177/0008125619864925

Hashim, S., Omar, M. K., Jalil, H. A., & Sharef, N. M. (2022). Trends on Technologies and Artificial Intelligence in Education for Personalized Learning: Systematic Literature Review. *International Journal of Academic Research in Progressive Education and Development*, *11*(1). https://doi.org/10.6007/ijarped/v11-i1/12230

Holicza, B., & Kiss, A. (2023). Predicting and Comparing Students' Online and Offline Academic Performance Using Machine Learning Algorithms. *Behav Sci (Basel)*, *13*(4), 289. https://doi.org/10.3390/bs13040289

Holmes, W., Bialik, M., & Fadel, C. (2023). Artificial intelligence in education. In *Data ethics : building trust : how digital technologies can serve humanity* (pp. 621–653). https://doi.org/10.58863/20.500.12424/4276068.

Horowitz, M., & Kahn, L. E. (2021). What influences attitudes about artificial intelligence adoption: Evidence from U.S. local officials. *PLOS ONE*, *16*(10), e0257732. https://doi.org/10.1371/journal.pone.0257732

Huang, J., Saleh, S., & Liu, Y. (2021). A Review on Artificial Intelligence in Education. *Academic Journal of Interdisciplinary Studies*, *10*(3), 206. https://doi.org/10.36941/ajis-2021-0077

Hwang, G. J., Xie, H., Wah, B. W., & Gašević, D. (2020). Vision, challenges, roles and research issues of Artificial Intelligence in Education. *Computers and Education: Artificial Intelligence*, *1*, 100001. https://doi.org/10.1016/j.caeai.2020.100001

ISTE. (2021). ISTE standards for educators: Implementing AI in education. Retrieved from https://www.iste.org/standards.

Javaid, M., Khan, S., Haleem, A., & Rab, S. (2022, November 8). *Adoption of modern technologies for implementing industry 4.0: an integrated MCDM approach*. Benchmarking: An International Journal. https://doi.org/10.1108/bij-01-2021-0017

Jiao, P., Ouyang, F., Zhang, Q., & Alavi, A. H. (2022). *Artificial intelligence-enabled prediction model of student academic performance in online engineering education*. *Artificial Intelligence Review*, *55*(8), 6321–6344. https://doi.org/10.1007/s10462-022-10155-y

Kelly, S., Kaye, S., & Oviedo-Trespalacios, O. (2023). What factors contribute to the acceptance of artificial intelligence? A systematic review. *Telematics and Informatics*, *77*, 101925. https://doi.org/10.1016/j.tele.2022.101925

Khan, I. A., Ahmad, A. L., Jabeur, N., & Mahdi, M. N. (2021). An artificial intelligence approach to monitor student performance and devise preventive measures. *Smart Learning Environments*, *8*(1). https://doi.org/10.1186/s40561-021-00161-y

Khare, K., Stewart, B. G., & Khare, A. (2018). Artificial Intelligence and the Student Experience: An Institutional Perspective. IAFOR Journal of Education, 6(3), 63–78. https://doi.org/10.22492/ije.6.3.04

Khor, E. T. (2014). An analysis of ODL student perception and adoption behavior using the technology acceptance model. *The International Review of Research in Open and Distributed Learning*, *15*(6). https://doi.org/10.19173/irrodl.v15i6.1732

Kim, J. H. (2019). Multicollinearity and misleading statistical results. *Korean Journal of Anesthesiology, 72*(6), 558–569. https://doi.org/10.4097/kja.19087

Koneru, I. (2017). Exploring Moodle Functionality for Managing Open Distance Learning E-Assessments. *The Turkish Online Journal of Distance Education*, 129–141. https://doi.org/10.17718/tojde.340402

Kuleto, V., Ilić, M., Dumangiu, M., Ranković, M., Martins, O. M. D., Păun, D., & Mihoreanu, L. (2021). Exploring Opportunities and Challenges of Artificial Intelligence and Machine Learning in Higher Education Institutions. *Sustainability*, *13*(18), 10424. https://doi.org/10.3390/su131810424

Kumar, S., & Choudhury, S. (2022). Gender and feminist considerations in artificial intelligence from a developing-world perspective, with India as a case study. *Humanities and Social Sciences Communications*, *9*(1). https://doi.org/10.1057/s41599-022-01043-5

Kurniawan, C., Kusumaningrum, S. R., Lam, K. T., & Surahman, E. (2022). Improving Language Teaching and Learning Process with Dual Coding Theory Approaches. *Jurnal Pendidikan: Teori, Penelitian, Dan Pengembangan*, *7*(8), 281. https://doi.org/10.17977/jptpp.v7i8.15313

Kurup, R., & Gupta, V. K. (2022). Factors Influencing the AI Adoption in Organizations. *Metamorphosis: A Journal of Management Research*, *21*(2), 129–139. https://doi.org/10.1177/09726225221124035

Lai, K. (2020). Fit Difference Between Nonnested Models Given Categorical Data: Measures and Estimation. *Structural Equation Modeling: A Multidisciplinary Journal*, 28, 99-120. https://doi.org/10.1080/10705511.2020.1763802

Lee, J. C., & Chen, X. (2022). Exploring users' adoption intentions in the evolution of artificial intelligence mobile banking applications: the intelligent and anthropomorphic perspectives. *International Journal of Bank Marketing*, *40*(4), 631–658. https://doi.org/10.1108/ijbm-08-2021-0394

Leeuwenberg, A. M., Van Smeden, M., Langendijk, J. A., Van Der Schaaf, A., Mauer, M. E., Moons, K. G. M., Reitsma, J. B., & Schuit, E. (2022). Performance of binary prediction models in high-correlation low-dimensional settings: a comparison of methods. Diagnostic and Prognostic Research, 6(1). https://doi.org/10.1186/s41512-021-00115-5.

Libasin, Z., Azudin, A. R., Idris, N. H., Rahman, M. N. A., & Umar, N. (2021). Comparison of Students' Academic Performance in Mathematics Course with Synchronous and Asynchronous Online Learning Environments during COVID-19 Crisis. *International Journal of Academic Research in Progressive Education and Development*, *10*(2). https://doi.org/10.6007/ijarped/v10-i2/10131

Li, H., Zhou, P., & Zhang, Z. (2010). An investigation into machine pattern recognition based on time-frequency image feature extraction using a support vector machine. *Proceedings of the Institution of Mechanical Engineers. Part C, Journal of Mechanical Engineering Science*, *224*(4), 981–994. https://doi.org/10.1243/09544062jmes1682

Lim, M. (2020). A study on the direction of technical Education in the age of artificial intelligence. *J. Korean Soc. Pract. Educ.*33, 81–102. doi: 10.24062/kpae.2020.33.4.81

Liu, X., & Huang, X. (2022). Design of Artificial Intelligence-Based English Network Teaching (AI-ENT) System. *Mathematical Problems in Engineering*, *2022*, 1–12. https://doi.org/10.1155/2022/1849430

Livieris, I. E., Δρακοπούλου, K., Tampakas, V., Mikropoulos, T. A., & Pintelas, P. E. (2018). Predicting secondary school students' performance utilizing a semi-supervised learning approach. *Journal of Educational Computing Research*, *57*(2), 448–470. https://doi.org/10.1177/0735633117752614

Lu, Y., Pian, Y., Chen, P., Meng, Q., & Cao, Y. (2021). RadarMath: An Intelligent Tutoring System for Math Education. *Proceedings of the . . . AAAI Conference on Artificial Intelligence*, *35*(18), 16087–16090. https://doi.org/10.1609/aaai.v35i18.18020

Makokotlela, M. V. (2022). Student Teachers' Experiences in Using Open Education Resource in the Open Distance Learning Context. *The Turkish Online Journal of Distance Education*, *23*(4), 108–120. https://doi.org/10.17718/tojde.1182763

Manhica, R., Santos, A., & Cravino, J. (2022). The use of artificial intelligence in learning management systems in the context of higher Education: Systematic literature review. *2022 17th Iberian Conference on Information Systems and Technologies (CISTI). https://doi.org/10.23919/cisti54924.2022.9820205*

Mathew, V. N., & Chung, E. (2021). University students' perspectives on open and distance learning (ODL) implementation amidst COVID-19. *Asian Journal of University Education*, *16*(4), 152. https://doi.org/10.24191/ajue.v16i4.11964

Mduma, N., Kalegele, K., & Machuve, D. (2019). A Survey of Machine Learning Approaches and Techniques for Student Dropout Prediction. *Data Science Journal*, *18*. https://doi.org/10.5334/dsj-2019-014

Msweli, P. (2012). Mapping the interplay between open distance learning and internationalisation principles. *The International Review of Research in Open and Distributed Learning*, *13*(3), 97. https://doi.org/10.19173/irrodl.v13i3.1182

Muhaimin, Habibi, A., Mukminin, A., Pratama, R., Asrial, & Harja, H. (2019). Predicting factors affecting intention to use web 2.0 in learning: evidence from science education. *Journal of Baltic Science Education*, *18*(4), 595–606. https://doi.org/10.33225/jbse/19.18.595

Nagy, M., & Molontay, R. (2023). Interpretable Dropout Prediction: Towards XAI-Based Personalized Intervention. *International Journal of Artificial Intelligence in Education*. https://doi.org/10.1007/s40593-023-00331-8

Namoun, A., & Alshanqiti, A. (2020). Predicting Student Performance Using Data Mining and Learning Analytics Techniques: A Systematic Literature Review. Applied Sciences, 11(1), 237. https://doi.org/10.3390/app11010237

Nguyen, A., Gardner, L. A., & Sheridan, D. (2020). A Design Methodology for Learning Analytics Information Systems: Informing Learning Analytics Development with Learning Design. In *Proceedings of the . . . Annual Hawaii International Conference on System Sciences*. https://doi.org/10.24251/hicss.2020.014

Nguyen, N. (2023). *The opportunities and Challenges of AI in Higher education*. Retrieved August 10, 2024, from https://feedbackfruits.com/blog/opportunities-and-challenges-of-ai-in-higher-education.

Nouraldeen, R. M. (2022). The impact of technology readiness and use perceptions on students' adoption of artificial intelligence: the moderating role of gender. *Development and Learning in Organizations*, *37*(3), 7–10. https://doi.org/10.1108/dlo-07-2022-0133

Nourani, V.; Gökçekuş, H.; Umar, I.K. (2020). Artificial intelligence-based ensemble model for prediction of vehicular traffic noise. Environ. Res. 180, 108852, doi: 10.1016/j.envres.2019.108852.

O'Dea, X., & O'Dea, M. (2023). Is Artificial Intelligence Really the Next Big Thing in Learning and Teaching in Higher Education? A Conceptual Paper. *Journal of University Teaching and Learning Practice*, *20*(5). https://doi.org/10.53761/1.20.5.05

Ogunsola-Bandele, M., & Kennepohl, D. (2022). "Gendered" Hardcore Sciences in a Male World-Across ODL and Non ODL Institutions. In *Tenth Pan-Commonwealth Forum on Open Learning*. https://doi.org/10.56059/pcf10.2776

Ojajuni, O., Ayeni, F., Akodu, O., Ekanoye, F., Adewole, S., Ayo, T., Misra, S., & Mbarika, V. (2021). Predicting student academic performance using machine learning. In *Lecture notes in computer science* (pp. 481–491). https://doi.org/10.1007/978-3-030-87013-3_36.

Ojo, A. I. (2017). Validation of the DeLone and McLean Information Systems Success Model. *Healthcare Informatics Research*, *23*(1), 60. https://doi.org/10.4258/hir.2017.23.1.60

Olivier, B. H. (2016). The Impact of Contact Sessions and Discussion Forums on the Academic Performance of Open Distance Learning Students. *The International Review of Research in Open and Distributed Learning*, *17*(6). https://doi.org/10.19173/irrodl.v17i6.2493

Onyema, E. M., Almuzaini, K. K., Onu, F. U., Verma, D., Gregory, U. S., Monika, P., & Afriyie, R. K. (2022). Prospects and Challenges of Using Machine Learning for Academic Forecasting. *Computational Intelligence and Neuroscience*, *2022*, 1–7. https://doi.org/10.1155/2022/5624475

Ouyang, F., Wu, M., Zheng, L., Zhang, L., & Jiao, P. (2023). Integration of artificial intelligence performance prediction and learning analytics to improve student learning in online engineering course. *International Journal of Educational Technology in Higher Education*, *20*(1). https://doi.org/10.1186/s41239-022-00372-4

Ouyang, F., Zheng, L., & Jiao, P. (2022). Artificial intelligence in online higher education: A systematic review of empirical research from 2011 to 2020. *Education and Information Technologies*, *27*(6), 7893–7925. https://doi.org/10.1007/s10639-022-10925-9

Oyedeji, A. B., Salami, A. M., Folorunsho, O., & Abolade, O. R. (2020). Analysis and Prediction of Student Academic Performance Using Machine Learning. *JITCE (Journal of Information Technology and Computer Engineering)*, *4*(01), 10–15. https://doi.org/10.25077/jitce.4.01.10-15.2020

Padilla, R. M. (2019). La llegada de la inteligencia artificial a la educación. *Revista De Investigación En Tecnologías De La Información*, *7*(14), 260–270. https://doi.org/10.36825/riti.07.14.022

Pandian, S. (2023). K-Fold Cross Validation Technique and its Essentials. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2022/02/k-fold-cross-validation-technique-and-its-essentials/

Petrova, D. I., & Bojikova, V. (2022). Development of two databases with comments in Bulgarian language and application of supervised learning approaches on them for comparative sentiment analysis. A brief overview. *Annual Journal of Technical University of Varna*, *6*(2), 57–62. https://doi.org/10.29114/ajtuv.vol6.iss2.261

Phua, P. L., Wong, S. L., & Abu, R. (2012). Factors Influencing the Behavioural Intention to use the Internet as a Teaching-Learning Tool in Home Economics. *Procedia - Social and Behavioral Sciences*, *59*, 180–187. https://doi.org/10.1016/j.sbspro.2012.09.263

Piccialli, V., & Sciandrone, M. (2022). Nonlinear optimization and support vector machines. *Annals of Operations Research*, *314*(1), 15–47. https://doi.org/10.1007/s10479-022-04655-x

Picciano, A. G. (2017). Theories and Frameworks for Online Education: Seeking an Integrated Model. *Online Learning*, *21*(3). https://doi.org/10.24059/olj.v21i3.1225

Pillai, R., & Sivathanu, B. (2020). Adoption of AI-based chatbots for hospitality and tourism. *International Journal of Contemporary Hospitality Management*, *32*(10), 3199–3226. https://doi.org/10.1108/ijchm-04-2020-0259

Pillai, R., & Sivathanu, B. (2020). Adoption of artificial intelligence (AI) for talent acquisition in IT/ITeS organizations. *Benchmarking: An International Journal*, *27*(9), 2599–2629. https://doi.org/10.1108/bij-04-2020-0186

Popenici, S., & Kerr, S. (2017). *Exploring the impact of artificial intelligence on teaching and learning in higher education*. Research and Practice in Technology Enhanced Learning, 12(1). https://doi.org/10.1186/s41039-017-0062-8

Rakya, Z. H. (2023). Exploring the Impact of Artificial Intelligence (AI) on Learner-Instructor Interaction in Online Learning (Literature Review). International Journal of Emerging Multidisciplinaries Computer Science & Artificial Intelligence, 2(1). https://doi.org/10.54938/ijemdcsai.2023.02.1.236.

Ramus, S. J., Elmasry, K., Luo, Z., Gammerman, A., Lu, K., Ayhan, A., Singh, N., McCluggage, W. G., Jacobs, I. J., Whittaker, J. C., & Gayther, S. A. (2023). Supplementary Data from Predicting Clinical Outcome in Patients Diagnosed with Synchronous Ovarian and Endometrial Cancer. *Supplementary Data From Predicting Clinical Outcome in Patients Diagnosed With Synchronous Ovarian and Endometrial Cancer*. https://doi.org/10.1158/1078-0432.22439479.v1

Reis, J., Santo, P. D. E., & Melão, N. (2020). Impact of artificial intelligence research on politics of the European Union Member States: The case study of Portugal. *Sustainability*, *12*(17), 6708. https://doi.org/10.3390/su12176708

Rifin, R., Kadiran, K. A., & Bakar, Z. A. (2022). Online Distance Learning for Electronic Design Subject at Diploma Level during Pandemic in Malaysia: A Case Study on Student Awareness and Perception. *International Journal of Academic Research in Progressive Education and Development*, 11(3). https://doi.org/10.6007/ijarped/v11-i3/14967

Rizwan, A., Iqbal, N., Ahmad, R., & Kim, D. (2021). WR-SVM Model Based on the Margin Radius Approach for Solving the Minimum Enclosing Ball Problem in Support Vector Machine Classification. *Applied Sciences*, *11*(10), 4657. https://doi.org/10.3390/app11104657

Roll, I., & Wylie, R. (2016). Towards learning analytics informed by and informing learning theory: Two case studies. Journal of Learning Analytics, 3(1), 23-46. https://doi.org/10.18608/jla.2016.31.3

Roll, I. & Wylie, R. (2016). Evolution and Revolution in Artificial Intelligence in Education. *International Journal of Artificial Intelligence in Education*, Volume 26, Issue 2, pp. 582-599. https://doi.org/10.1007/s40593-016-0110-3

Rukhsar, S., Tiwari, A. K., & Panda, S. (2022). Deep Optimized Electrodes and Frequency Bands in the Phase Space for Identification of Seizures. In *2022 IEEE 19th India Council International Conference (INDICON)*. https://doi.org/10.1109/indicon56171.2022.10040195

Sabeh, H. N., Husin, M. H., Kee, D. M. H., Baharudin, A. S., & Abdullah, R. (2021). A Systematic Review of the DeLone and McLean Model of Information Systems Success in an E-Learning Context (2010–2020). *IEEE Access*, *9*, 81210–81235. https://doi.org/10.1109/access.2021.3084815

Saini, R. (2022). Integrating Vegetation Indices and Spectral Features for Vegetation Mapping from Multispectral Satellite Imagery Using AdaBoost and Random Forest Machine Learning Classifiers. *Geomatics and Environmental Engineering*, *17*(1), 57–74. https://doi.org/10.7494/geom.2023.17.1.57

Sakibayev, S., Sakibayev, R., & Sakibayeva, B. (2019). The educational impact of using mobile technology in a database course in college. *Interactive Technology and Smart Education*, *16*(4), 363–380. https://doi.org/10.1108/itse-12-2018-0103

Sallam, K. M., Elsayed, S., Sarker, R. A., & Essam, D. (2020). Landscape-assisted multi-operator differential evolution for solving constrained optimization problems. *Expert Systems With Applications*, *162*, 113033. https://doi.org/10.1016/j.eswa.2019.113033

Salmer'on, R., Garc'ia, C., & Garc'ia, J. (2020). Overcoming the inconsistences of the variance inflation factor: a redefined VIF and a test to detect statistical troubling multicollinearity. *arXiv: Methodology*. Consensus. Retrieved February 11, 2024, from https://consensus.app/papers/overcoming-inconsistences-variance-inflation-factor-salmeron/2701f6aa76e8527c87c9f9ed439e28d7/

Sandra, L., Lumbangaol, F., & Matsuo, T. (2021). Machine Learning Algorithm to Predict Student's Performance: A Systematic Literature Review. *TEM Journal*, 1919–1927. https://doi.org/10.18421/tem104-56

Samsudin, N. a. M., Shaharudin, S. M., Sulaiman, N. a. F., Smail, S. I., Mohamed, N. S., & Husin, N. H. M. (2022). Prediction of Student's Academic Performance during Online Learning Based on Regression in Support Vector Machine. *International Journal of Information and Education Technology*, *12*(12), 1431–1435. https://doi.org/10.18178/ijiet.2022.12.12.1768

Seo, K. W., Tang, J., Roll, I., Fels, S., & Yoon, D. (2021). The impact of artificial intelligence on learner–instructor interaction in online learning. International Journal of Educational Technology in Higher Education, 18(1). https://doi.org/10.1186/s41239-021-00292-9

Shen, L., Chen, I. A., Grey, A. K. M., & Su, A. (2021). Teaching and Learning With Artificial Intelligence. In *Advances in educational technologies and instructional design book series* (pp. 73–98). IGI Global. https://doi.org/10.4018/978-1-7998-4763-2.ch005

Shen, M., Russek-Cohen, E., & Slud, E. V. (2014). Exact calculation of power and sample size in bioequivalence studies using two one-sided tests. *Pharmaceutical Statistics*, *14*(2), 95–101. https://doi.org/10.1002/pst.1666

Shen, Y. (2023). Academic Performance in Transition to Online Distance Learning: An Assessment from Prior Academic Performance Across Subjects. *Journal of Education, Humanities and Social Sciences*, *8*, 634–641. https://doi.org/10.54097/ehss.v8i.4320

Shi, D., Distefano, C., Maydeu-Olivares, A., & Lee, T. (2021). Evaluating SEM Model Fit with Small Degrees of Freedom. *Multivariate Behavioral Research*, 57, 179-207. https://doi.org/10.1080/00273171.2020.1868965

Shi, D., & Maydeu-Olivares, A. (2020). The Effect of Estimation Methods on SEM Fit Indices. *Educational and Psychological Measurement*, 80, 421-445. https://doi.org/10.1177/0013164419885164

Singam, A. K., Lövström, B., & Kulesza, W. J. (2023). Comparative Studies of Unsupervised and Supervised Learning Methods based on Multimedia Applications. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2303.02446

Strzelecki, A. (2023). Students' Acceptance of CHATGPT in Higher Education: An Extended Unified Theory of Acceptance and Use of Technology. *Innovative Higher Education*. https://doi.org/10.1007/s10755-023-09686-1.

Sun, Y. (2016). *The improved particle swarm breaker fault status parameter optimization of SVM classification*. https://doi.org/10.2991/icmmita-16.2016.195

Squicciarini, M., Borgonovi, F., Andrieu, E. and liebender, A. (2020). The role of Education and

skills in bridging the digital gender divide evidence from APEC economies. Retrieved from https://www.researchgate.net/publication/338920260_the_role_of_education_and_skills_in_bridging_the_digital_gender_divide_evidence_from_apec_economies

Tait, A. R. (2014). From Place to Virtual Space: Reconfiguring Student Support for Distance and E-Learning in the Digital Age. *Open Praxis*, *6*(1), 5. https://doi.org/10.5944/openpraxis.6.1.102

Tait, H., & Godfrey, H. (2001). Enhancing the Student Experience for Direct Entrants to the Penultimate Year of Undergraduate Degree Programmes. Journal of Further and Higher Education, 25(2), 259–265. https://doi.org/10.1080/03098770120050918

Tanjga, M. (2023). E-learning and the Use of AI: A Review of Current Practices and Future Directions. *Qeios*. https://doi.org/10.32388/ap0208.2

Tanveer, M., Hassan, S., & Bhaumik, A. (2020). Academic Policy Regarding Sustainability and Artificial Intelligence (AI). Sustainability, 12(22), 9435. https://doi.org/10.3390/su12229435

Tiwari, R. (2023). The integration of AI and machine learning in education and its potential to personalize and improve student learning experiences. *Indian Scientific Journal of Research in Engineering and Management*, *07*(02). https://doi.org/10.55041/ijsrem17645

Togaibayeva, A., Ramazanova, D., Yessengulova, M., Yergazina, A., Nurlin, A., & Shokanov, R. (2022). Effect of mobile learning on students' satisfaction, perceived usefulness, and academic performance when learning a foreign language. *Frontiers in Education*, *7*. https://doi.org/10.3389/feduc.2022.946102

Tomašević, N., Gvozdenovic, N., & Vraneš, S. (2020). An overview and comparison of supervised data mining techniques for student exam performance prediction. *Computers & Education*, *143*, 103676. https://doi.org/10.1016/j.compedu.2019.103676

Top 5 Challenges of Adopting AI in Education. (2021). Artificial Intelligence Board of America. Retrieved January 3, 2023, from https://www.artiba.org/blog/top-5-challenges-of-adopting-ai-in-education

Toplic, L. (2021). *If AI is the future, gender equity is essential*. NetHope. Retrieved December 26, 2022, from https://nethope.org/articles/if-ai-is-the-future-gender-equity-is-essential/

UNESCO. (2019). Artificial Intelligence in Education: Challenges and Opportunities for Sustainable Development. (2019). *United Nations Educational, Scientific and Cultural Organization*.

UNESCO. (2022). International Forum on AI and Education: Ensuring AI as a Common Good to Transform Education, 7-8 December 2021; synthesis report. Retrieved October 16, 2022, from https://unesdoc.unesco.org/ark:/48223/pf0000381226

Uunona, G. N., & Goosen, L. (2023). Leveraging Ethical Standards in Artificial Intelligence Technologies. In *Advances in medical education, research, and ethics (AMERE) book series* (pp. 310–330). CRC Press. https://doi.org/10.4018/978-1-6684-7164-7.ch014

Valentin, Y., Fail, G., & Pavel, U. (2022). Shapley values to explain machine learning models of school student's academic performance during COVID-19. *3C TIC*, *11*(2), 136–144. https://doi.org/10.17993/3ctic.2022.112.136-144

Vasileiou, K., Barnett, J., Thorpe, S. J., & Young, T. (2018). *Characterising and justifying sample size sufficiency in interview-based studies: systematic analysis of qualitative health research over a 15-year period*. BMC Medical Research Methodology. https://doi.org/10.1186/s12874-018-0594-7

Wang, H., Shao, Y., Zhou, S., Zhang, C., & Xiu, N. (2021). Support Vector Machine Classifier via $L_{0/1}$ Soft-Margin Loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1. https://doi.org/10.1109/tpami.2021.3092177

Wang, Youmei, Liu, C., & Tu, Y.-F. (2021). Factors Affecting the Adoption of AI-Based Applications in Higher Education. Retrieved from https://www.jstor.org/stable/e27032850

Wang, Y., Liu, C., Tu, Y.F. (2021). Factors Affecting the Adoption of AI-Based Applications in Higher Education: An Analysis of Teachers' Perspectives Using Structural Equation Modelling. *Educational Technology & Society*, *24* (3), 116–129.
Environ. Res. 180, 108852, doi: 10.1016/j.envres.2019.108852.

Xia, Y., & Yang, Y. (2018). RMSEA, CFI, and TLI in structural equation modeling with ordered categorical data: The story they tell depends on the estimation methods. *Behavior Research Methods*, 51, 409-428. https://doi.org/10.3758/s13428-018-1055-2

Xiao, S., Shanthini, A., & Thilak, D. (2021). Instructor Performance Prediction Model Using Artificial Intelligence for Higher Education Systems. *Journal of Interconnection Networks*, *22*(Supp03). https://doi.org/10.1142/s0219265921440035

Xu, Z. (2024). AI in education: Enhancing learning experiences and student outcomes. *Applied and Computational Engineering*, *51*(1), 104–111. https://doi.org/10.54254/2755-2721/51/20241187.

Yağcı, M. (2022). Educational data mining: prediction of students' academic performance using machine learning algorithms. *Smart Learning Environments*, *9*(1). https://doi.org/10.1186/s40561-022-00192-z

Yakubu, M. N., & Dasuki, S. I. (2018). Factors affecting the adoption of e-learning technologies among higher education students in Nigeria. *Information Development*, *35*(3), 492–502. https://doi.org/10.1177/0266666918765907

Yang, X., Zhang, Y., & Li, X. (2023). Abnormal noise identification of engines based on Wavelet transform and bispectrum analysis. *Research Square (Research Square)*. https://doi.org/10.21203/rs.3.rs-2688747/v1

Yannier, N., Hudson, S. E., Koedinger, K. R., Hirsh-Pasek, K., Golinkoff, R. M., Munakata, Y., . . . Brownell, S. E. (2021). Active learning: "Hands-on" meets "minds-on." *Science*, *374*(6563), 26–30. https://doi.org/10.1126/science.abj9957

Zhang, Y., Yun, Y., An, R., Cui, J., Dai, H., & Shang, X. (2021). Educational Data Mining Techniques for Student Performance Prediction: method review and comparison analysis. *Frontiers in Psychology*, *12*. https://doi.org/10.3389/fpsyg.2021.698490.

Zhu, Z. (2021, December 15). Explain Support Vector Machines in Mathematic Details. *Medium*. Retrieved May 16, 2023, from https://towardsdatascience.com

Zhu, Z., Liu, Q., & Li, H. (2018). The application of artificial intelligence in open and distance learning: A review. International Journal of Emerging Technologies in Learning, 13(7), 114-126. https://doi.org/10.3991/ijet.v13i07.8356

Appendix A: The Pseudocode for the Improved SVM (Improved VIF Optimization)

```
START
// Step 1: Load Data
```

LOAD dataset using pandas

// Step 2: Ordinal Encoding
FOR each Likert scale column in dataset
APPLY ordinal encoding
// Step 3: Handle Missing Data
FOR each column in dataset
CHECK if missing data exists
IF missing data exists THEN
CALCULATE average of non-missing values in the same column (construct)
REPLACE missing data with average value

// Step 4: Compute Composite Scores
FOR each construct in the dataset
CALCULATE the mean of the associated items
STORE the mean value as the composite score of the construct

// Step 5: Verify Internal Consistency
FOR each construct in dataset
CALCULATE Cronbach's Alpha
IF Cronbach's Alpha is less than acceptable value THEN
FLAG the construct for review

// Step 6: Data Preparation for SVM
CREATE a new dataset with AI related constructs as independent variables
SET Students' Academic Performance as Target variable

// Step 7: Train-Test Split
SPLIT the dataset into a training set and a test set

// Step 8: Train SVM Model
INITIALIZE SVM model with parameters
FIT the model with the training set

// Step 9: Evaluate the SVM Model
PREDICT the target variable for the test set using the trained model
CALCULATE evaluation metrics (MAE, MSE, MAPE, RMSE, NMSE)

END

Appendix B: The Questionnaire used for data collection

QUESTIONNAIRE ON THE IMPACT OF ARTIFICIAL INTELLIGENCE ON ACADEMIC PERFORMANCE IN OPEN AND DISTANCE LEARNING

INTRODUCTION

Good day, Sir/Madam,

Thank you for considering participating in this study, which aims to develop a process framework for predicting the impact of artificial intelligence adoption on students' academic performance in Open and Distance Learning (ODL) using a support vector machine.

- **Participation:** Your participation is voluntary, and you may withdraw at any time.
- **Procedure:** The questionnaire will take approximately 15-20 minutes to complete.
- **Confidentiality:** All responses are confidential and anonymized.
- **Risks and Benefits:** There are no known risks associated with participating in this study.
- **Consent:** By proceeding, you voluntarily agree to participate and confirm you are above 18 years old.

For any queries, please get in touch with the researcher. Your participation is highly valued.

.

Warm regards,

Muyideen Adewale
PhD Candidate in Artificial Intelligence,
Africa Centre of Excellence on Technology Enhanced Learning.
Phone: +14379944562
Email: Ace22140007@noun.edu.ng; mdadewale@gmail.com

Section A: Demographics

1. Please provide your age group: [Below 20] [20-29] [30-39] [40-49] [50 and above]

2. Please indicate your gender: [Male] [Female] [Prefer not to say]
3. Please provide your geographical location (country): [Canada] [Nigeria]
4. What is your field of study? [Computer Science] [Information Technology] [Business Administration] [Marketing] [Engineering] [Natural Sciences] [Social Sciences] [Humanities] [Arts & Design] [Education] [Health & Medicine] [Agriculture & Environmental Sciences] [Mathematics & Statistics] [Physical Sciences] [Biological & Life Sciences] [Law & Legal Studies] [Journalism & Media Studies] [Philosophy & Theology] [Psychology] [Other - Please Specify]

Please answer the following questions after using the application. The information below provides the code and meaning for each option to be ticked.

1- Strongly Disagree 2- Disagree 3-Neither Agree nor Disagree 4- Agree 5- Strongly Agree

Section B: AI Alignment and Relevance (AAR)

1. I feel that the AI-based Moodle platform used in my course aligns well with my learning needs and objectives.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. The AI-based Moodle platform implemented in my institution aligns with its educational goals and values.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. The use of AI-based Moodle platform features makes my course content more relevant.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. Using the AI-based Moodle platform in my course positively impacts my attitude towards technology in education.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section C: Comparative Advantage of AI (CAAI)

1. Learning with the AI-based Moodle platform is more effective than traditional educational methods.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. The AI-based Moodle platform features provide significant advantages to my learning process compared to traditional methods.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. Learning with the AI-based Moodle platform is more efficient in terms of time and resource utilization.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. The AI-based Moodle platform enhances the effectiveness of my learning outcomes compared to traditional methods.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section D: Ease and Enjoyment of Use (EEU)

1. I find it easy to use the AI-based Moodle platform for learning in my course.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. My experience interacting with the AI-based Moodle platform in my course is enjoyable.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. Learning with the AI-based Moodle platform is intuitive and user-friendly.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. The use of the AI-based Moodle platform in my course is engaging and motivating.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section E: AI Readiness and Facilitating Conditions (ARFC)

1. I feel well-prepared to use the AI-based Moodle platform in my learning.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. My institution is well-prepared for adopting and implementing the AI-based Moodle platform.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. I receive substantial support (technical, learning resources, etc.) in using the AI-based Moodle platform for learning.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. The conditions in my institution facilitate the effective use of the AI-based Moodle platform for learning.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section F: AI-induced Learning Anxiety (AILA)

1. I often feel anxious or stressed about using the AI-based Moodle platform in my course.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. I feel worried about relying on the AI-based Moodle platform for learning.

1	2	3	4	5

Strongly Disagree ☐ ☐ ☐ ☐ ☐ Strongly Agree

3. I often feel overwhelmed by the complexity of the AI-based Moodle platform used in my course.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. I worry that errors or problems in the AI-based Moodle platform could negatively impact my learning outcomes.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section G: Interactive Capability (IC)

1. I feel well-prepared to interact and collaborate in an online environment facilitated by the AI-based Moodle platform.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. The AI-based Moodle platform has enhanced my ability to interact with teachers and peers.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. The use of the AI-based Moodle platform has positively impacted my collaboration in group projects or activities.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. The AI-based Moodle platform facilitates effective communication in my learning environment.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section H: Knowledge Absorption and User Satisfaction (KAUS)

1. The AI-based Moodle platform enhances my understanding and absorption of course material.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. I am satisfied with my learning outcomes due to the use of the AI-based Moodle platform.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. The AI-based Moodle platform often aids in clarifying complex course material or concepts.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. The use of the AI-based Moodle platform improves my satisfaction with the learning experience.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section I: Systems Quality and Social Influence (SQSI)

1. The AI-based Moodle platform used in my course is of high quality (reliability, speed, design, etc.).

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. The views of my peers significantly influence my usage of the AI-based Moodle platform in my course.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. Social media, discussions with peers, or instructors' opinions have a strong impact on my acceptance and use of the AI-based Moodle platform.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. High-quality AI systems enhance their acceptance and use among my peers.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Section J: Students' Academic Performance

1. I believe that using AI tools like the AI-based Moodle platform has improved my academic performance.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

2. AI in online learning has helped me better understand the course materials.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

3. AI tools like the AI-based Moodle platform have contributed to better grades in my courses.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

4. How would you classify your Cumulative Grade Point Average (CGPA) on a scale of 5? Please select the range that applies to your academic performance. [First Class Honors (4.50 - 5.00)] [Second Class Honors, Upper Division (3.50 - 4.49)] [Second Class Honors, Lower Division (2.50 - 3.49)] [Third Class Honors (1.50 - 2.49)] [Pass (1.00 - 1.49)]

Appendix C: The University Ethics Committee Approval

Gmail MD Adewale <mdadewale@gmail.com>

Request for Ethics Committee Approval of PhD Research Questionnaire

Grace JOKTHAN <gjokthan@noun.edu.ng> Fri, Apr 26, 2024 at 2:44 AM
To: MD Adewale <mdadewale@gmail.com>
Cc: Africa Centre of Excellence on Technology Enhanced Learning <acetel@noun.edu.ng>, acetelregistry@gmail.com, ACETEL Registry <acetelregistry@noun.edu.ng>, Gregory ONWODI <gonwodi@noun.edu.ng>, researchadministration@noun.edu.ng, Ambrose Azeta <azetaambrose@gmail.com>

Hello Adewale,
Hope this mail finds you well.
The University Ethics Committee has approved your request and the secretary is expected to issue the approval letter. We will follow up and forward the memo to you today or early next week.
Congratulations
The Centre is following up on the request for sponsorship as it requires the Vice Chancellor's approval. We are hopeful that the request will come through.
Thank you
[Quoted text hidden]

--
Prof. Grace Jokthan
Director, Africa Centre of Excellence on Technology Enhanced Learning (ACETEL)
National Open University of Nigeria (NOUN)
University Village, Plot 91, Cadastral Zone,
Nnamdi Azikiwe Expressway Jabi-Abuja
+2348(0)8182972097, +234(0)8166298072

DEVELOPMENT OF A FRAMEWORK FOR AREA-BASED EXPLOSIVE TRACE DETECTION USING DEEP TRANSFER LEARNING

BY

OHEMU, MONDAY FREDRICK (ACE21140003)

AFRICAN CENTRE FOR EXCELLENCE ON TECHNOLOGY ENHANCED LEARNING (ACETEL), NATIONAL OPEN UNIVERSITY OF NIGERIA, ABUJA

FEBRUARY, 2024

DEVELOPMENT OF A FRAMEWORK FOR AREA-BASED EXPLOSIVE TRACE DETECTION USING DEEP TRANSFER LEARNING

By

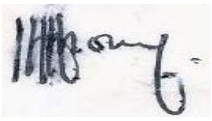
Monday Fredrick OHEMU

ACE21140003

A Thesis Submitted in Partial Fulfilment of the Requirements for the Award of the Degree of Doctor of Philosophy (Ph.D.) in Artificial Intelligence at the Africa Centre of Excellence on Technology Enhanced Learning , National Open University of Nigeria

DECLARATION

I, **Monday Fredrick OHEMU (ACE21140003)** hereby declare that the project work entitled: **DEVELOPMENT OF A FRAMEWORK FOR AREA-BASED EXPLOSIVE TRACE DETECTION USING DEEP TRANSFER LEARNING** is a record of an original work done by me, as a result of my research effort carried out in the Africa Centre of Excellence on Technology Enhanced Learning, National Open University of Nigeria under the supervision of

____  ____19/02/2024___

Student's Signature & Date

CERTIFICATION / APPROVAL

This is to certify that this study was carried out by **Monday Fredrick OHEMU** with Matric Number **ACE21140003** at the **Africa Centre of Excellence on Technology Enhanced Learning**, National Open University of Nigeria, under my supervision.

20 February 2024

Prof. Ambrose A. Azeta …….

Main Supervisor Signature & Date

Prof. Adeyanju Ibrahim **20-FEB-2024**

……

Co-Supervisor Signature & Date

Mr. Nuradeen Maidoki …….

Industry Supervisor Signature & Date

Prof. Grace Jokthan ……

Centre Director Signature & Date

Prof. Greg Onwodi ……

Programme Coordinator Signature & Date

…… ……

External Examiner Signature & Date

DEDICATION

This thesis is dedicated to Almighty God for His mercy and love that has brought me to this level and to late Mr. Raymond Akor for his encouragement and support in my academic pursuit

ACKNOWLEDGEMENTS

I would like to extend my deepest appreciation to my esteemed supervisors, Prof. Ambrose A. Azeta and Prof. Ibrahim Adeyanju for entrusting me with the pursuit of my PhD in a subject that is particularly close to my heart. Without their unwavering guidance and support, my journey through this challenging academic pursuit would have been futile. I am eternally grateful for their encouragement, and commitment to my success. Their mentorship has been instrumental in helping me achieve my academic dreams, and I will always treasure their invaluable contribution to my academic and professional development.

My gratitude also goes to the staff and management of ACETEL, particularly the Centre Director, Prof. Grace Jokthan, AI program Coordinator Prof. Greg Onwodi, the ICT Director, Ibrahim Abubakar, the head of Admin, Udochukwu and to all my lecturers who impacted me to make this programme possible.

My lovely wife stood by my side throughout this journey and believed in me, Thank you. I say thank you to all my family members, Augustine, Francis, Edward, Regina and the Akors. I appreciate the encouragement friends like Moses Anthony, Dr. Obasi Chukwuemeka, Francis Sylvanus and Senior Friends like Dr. Julius Olajire, Dr. Innocent Oguche.

Finally, I will like to thank my class mates, 2020/2021Pioneer set of PhD Artificial Intelligent ACETEL-NOUN, Rexcharle, Dere, Bayole, Ganfather, Keture, it was wonderful working with you people.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
CNN	Convolution Neural Network
CV	Computer Vision
DCNN	Deep Convolution Neural Network
DTLETD	Deep Transfer Learning for Explosive Trace Detection
ETD	Explosive Trace Detection
DNN	Deep Neural Network
GC	Gas Chromatography
HMX	Octahydro- 1,3,5,7 –tetranitro- 1,3,5,7 –tetrazocine
IED	Improvised Explosive Device
IOT	Internet of Things
KNN	K-Nearest Neighbor
MS	Mass Spectrometry
NLP	Natural Language Processing
RDX	1, 3, 5-triazacyclohexane
TNT	2,4,6- Trinitrotoluene
WSN	Wireless Sensor Network

ABSTRACT

Terrorism and the proliferation of explosives has caused serious damage in public places and has become an issue of serious security concern across the globe. Most public places such as airports, trains stations, government institutions and facilities are being targeted, thereby endangering the safety of people's life and facilities. It is essential to protect these target areas from explosions and terrorist attacks, without necessarily exposing human security personnel to such danger. In an attempt to solve the aforementioned problem several approaches such as animals have been engaged. However, machine learning models have been proven to provide better solutions. The accuracy of machine learning model depends on large volume of data although some specific type of training has its own setbacks, since obtaining large volume of data for such training may be cumbersome. The need to develop systems that can easily adapt with less data and little training knowledge has become inevitable. The focus of this work is to develop a framework for area-based explosive trace detection using deep transfer learning. The model used was adapted from deep learning technology trained with large explosive trace data set that were collected from sensor network. The dataset was converted to a 2D data using serial data to image generator. The model was developed from a base model known as GasNet and classified explosive gas within an area base on the concentration of Carbon (C), Hydrogen (H), Oxygen (O), and Nitrogen (N) gases and was able to classify the gas combinations as either explosive or not. The developed model called deep transfer learning for explosive trace detection (DTLETD) was tested and validated using 10% of the explosive trace dataset with the transfer learning model taking less time of about 92 seconds to train against a training time of about 1287 seconds for Convolutional Neural Network (CNN) base model. The transfer learning model converged faster with nearly zero losses for both training and validation. The model also recorded an accuracy of 99.7%, with an average AUC value of about 0.89. The outcome has a precession of 96% against 98.2% accuracy and AUC of 1 that was recorded with the base model. The system was able to adapt with good performance to the new data within little time using few datasets. This research was able to achieve its objective of developing a framework for area base explosive trace and was able to improve the accuracy of explosive trace detection through the development of machine learning based model that utilizes the Deep Transfer Learning (DTL) approach.

CHAPTER ONE
INTRODUCTION

1.1. BACKGROUND OF THE STUDY

Attacks on people and sensitive places in the form of terrorism has become a global challenge that is making organizations such as academic institutions, security agencies and the government to do whatever it takes to secure people and essential infrastructures. Recently, explosive-based attacks on essential equipment, students, personnel and government have become rampant because explosive form of weapons is easy to manufacture and deploy and can cause serious destruction (Al-mousawi & Al-mousawi, 2019). This has made development of various kinds of explosives for destroying innocent people and properties very common. The area of interest includes learning institutions, airport, government properties and military base which can be monitored through sensor network. The sensor network which comprises different types of sensors is designed and deployed continuously to detect and identify explosive traces within specific treat locations in an environment. Trace elements, compounds, or chemical residues associated with explosives, such as TNT (trinitrotoluene), RDX (hexahydro-1,3,5-trinitro-1,3,5-triazine), known as the Royal Demolition Explosive or PETN (pentaerythritol tetranitrate) can be detected. This information collected in real time by the sensor network can be process either by the sensor note or by a remote server using advanced algorithm for data analysis. This has led to the development of Artificial intelligent (AI) based system to accurately detect explosives before causing havoc in an environment. This will eliminate the manual ways of screening by human security system to monitor and secure target environment that further expose human being to potential attacks in volatile areas (Wongwattanaporn, 2021). This work focuses on how to effectively use AI based technology to secure and effectively monitor target environment identified as terrorist potential attack area using sensor networks system.

Two known methods that have been deployed in explosive detection are bulk explosive detection method and trace explosive detection method. Meanwhile, the bulk explosive method uses method such as X-rays and other electromagnetic imaging method such as the recent computer tomography. This method is based on visual, optical and thermal characteristics of explosive substances that requires advance image processing applications for implementation of thermo-optical sensors in achieving better result. Contrary to the bulk explosive detection methods, the trace explosive detection approach is based on chemical property traces of

explosive materials (Kishore et al., 2019) with most high explosives having the general formula of C_a H_b N_d O_k, where the subscript a, b, d, k are numbers of atom associated with each element. The sample containing oxidizer (O) and the fuel (C, H) of different degree before explosive will be formed, as in the case of RDX (Cyclotrimethylenetrinitramine, C_3H_6 N_6O_6) (Pai, Peng; Xiaojin, Zhao;Xiaofang, Pan;Wenbin, 2018) (Royal Society of Chemistry, 2011). Since each chemical has distinctive characteristics that may alter its environmental composition by altering some chemical or physical characteristic in the environment. Special sensors are designed to observe these changes often associated to wireless sensor network (WSN) system applied for the detection of explosives substances. The main properties considered for explosive detection procedure includes the chemical characteristic, mechanical nature and physical nature of the material. The chemical nature of the explosive substance does change the chemical nature of the material in its environment, resulting in alteration in environmental composition of the surrounding. Sensors that have either chemical or physical ability to detect these changes are usually used to respond to these chemical and mechanical changes. It should be noted that the mechanical nature of substances are physically related to the motion of substances such as pressure and speed of these explosive substances and are easily detected using mechanical sensors (Al-mousawi & Al-mousawi, 2019)

The focus of current technological advancement in the study area is towards early detection of explosive and the trace detection method which is a faster approach, since at certain pressure and temperature, solid and liquid substances yield vapours that depends on the quantity of that vapours to produce a volatility of a substance. A suitable approach of sampling of the gas and analytical methods in the presence of particular substance makes early detection possible depending on the level of concentration of the substance. The possibility to detect the vapours of interest is directly determined by how volatile the substance maybe (Wasilewski & Gębicki, 2021). Early detection of explosive substance that could possibly be used is beginning to take preference in detection approach. In this approach used for early detection, substances that could be used to manufacture explosives with emphasis mainly on trace of such substances within the areas that are meant to be explosive free are been explored. It has been established that IED are types of bonds commonly used by terrorist and they are made up of certain chemicals (Wang et al., 2019).

1.2. Problem Statement

Terrorism and the proliferation of explosives has caused serious damage in public places and has become an issue of serious security concern across the world (Obasi et al., 2023), with most public places such as airports, trains stations, government institutions and facilities being targeted, thereby endangering the safety of people and facilities. How can these target areas be safe from explosions and terrorists attack without necessarily exposing human security personnel to such danger? In an attempt to solve the aforementioned problem several approaches have been used such as animal Chuen *et al.,* (2020), chemicals ions Adegoke & Nic Daeid, (2021) Hao *et al.,* (2022), mechanical devices, X and gamma rays, neutrons(Al-mousawi & Al-mousawi, 2019) and electronics nose-based Liu et al., (2019). These methods have been developed to be intelligent and effective in monitoring wild area and have a short range of detection. Also, these approaches sometimes result in bulky instrument for detection that become too noticeable, such visibility could make terrorists to attempt to beat the solutions provided (Wongwattanaporn, 2021). Since the explosive trace properties cannot be identified or detected by the human senses, an artificial intelligent based system can be deployed to detect the presence of explosive trace within an environment with high accurately. Sensor network are able to sense explosive trace properties and represent same as numeric values that can serve as input to another system (Al-Mousawi & Al-Hassani, 2018). This is based on the ability of the sensor to track the chemical and physical characteristics of the traces that these explosives emit to the surrounding environment. These substances can be traced by Artificial Intelligent (AI) based wireless sensor network (WSN) that are highly sensitive. The accuracy of the sensor network is very paramount to have a sensitive robust system. Explosive trace substance data are scarce as a result of privacy and ethical concern, which will make deep transfer learning (DTL) approach one of the best techniques to solve this study's problem statement.

The DTL model was based on sensor output labeled from a time-series data collected other source trained for deep neural network (DNN). The DTL was used to learn the general behavior of time-series data before transferring it to another DNN that is developed for the purposed of explosive trace detection. Peculiar features of explosive traces are usually extracted through the normal traditional method of feature extraction which has now been overshadowed by an automated approach of feature learning such as the DNN (Huang *et al.*, 2020; Fisher et al., 2020).

Various types of sensors network can be utilized in collecting different gaseous component together with the physical properties of the environment. In this research, sensor-based data of explosive trace were collected at various location within an environment, and used to developed the machine learning model based on DTL.

In order to realize the purpose of this research, there was need to introduce an Artificial Intelligent model for securing a learning environment with high selectivity and accuracy with the capability to adapt speedily with limited explosive trace data (Yaqoob & Younis, 2021), as a result of restriction in explosive chemical production (Omijeh & Okemeka Machiavelli, 2019). Several others such as traditional machine learning models used to solve similar problem can only detect explosive that have been trained and the deep learning model requires much time and large volume of dataset for training (Wang et al., 2022). A model called Deep transfer learning for explosive trace detection (DTLETD) that can use limited explosive data was designed in this work to accurately detect explosive trace among other chemicals in a learning environment with a reduced training time

.

1.3. Aim and Objectives

The aim of this study is to develop a model for area-based explosive trace detection using deep transfer learning. The specific objectives of the study are:

I. To carry out requirements elicitation of area-based explosive trace detection
II. To design a framework for the detection of explosive traces using Deep Transfer Learning algorithm
III. To implement a prototype of the framework for area-based explosive trace detection using deep transfer learning
IV. To evaluate and benchmark the performance of the developed model relative to a current state-of-the-art model

1.4. Research Methodology

Table 1.1 shows the mapping of the objectives to the research methodology with corresponding question to be addressed. The requirement for explosive detection was established through literature and the model for explosive trace detection using deep transfer learning was designed using appropriate design tool. The proposed model was implemented using python and its Libraries. The last objective to evaluate the deep transfer learning model was achieved using confusion matrix.

Table 1.1: The objectives, research methodology, and research questions mapping.

SN	Research objectives	Research Methodology	Research Questions
I	To Carryout requirements elicitation of area-based explosive trace detection	-Literature review -Primary and secondary sources of data.	What are the requirements of area-based explosive trace detection?
II	To design a framework for the detection of explosive traces using Deep Transfer Learning algorithm	-Use of schematic diagram, Visio, draw.io Unified Modelling Language (UML)	How can a framework for the detection of explosive traces using Deep Transfer Learning be designed?
III	To implement the framework for area-based explosive trace detection using deep transfer learning	-Use of Python - Anaconda.Navigator, Jupiter Notebook. -Support Vector Machine. -Libraries: Pandas, Numpy, Sklearn, matplotlib, imblearn, etc.	How can a framework for the detection of explosive traces using Deep Transfer Learning be implemented?
IV	To test and evaluate the performance of the transfer learning model and benchmark with other state-of-the-art model like SVM, KNN, CNN, ImageNet and AlexNet	-Standard machine learning evaluation techniques (Recall, Precision, F1-Score, Accuracy),	How can machine learning models of Deep transfer learning be evaluated to determine the level of accuracy?

1.5. Scope of the Study

This research proposes to detect explosives within an area by using the common trace of explosive material. A deep learning model was be developed to detect the presence of explosive trace from the limited available data collected from sensor array network. This model expected to be accurate, fast in detection and at the same time light weighted to be able to run on edge device. The system was not considered for the bulk type of explosives, however the model can be train to work on bulk-based sensors. The system will be validated using dataset generated from a simulated setup to determine the performance of the developed model. The work will not consider sensor and wireless sensor placement. The developed model is not expected to be used on edge devices, since it's outside the scope of this work.

1.6.Significance of the Study

Explosive-based terrorism has become a known means of carrying out attacks on public places, such as learning environment, train station, airports and sensitive facilities which has caused a lot of damages to lives and properties. The reason for this rise in explosive attacks is because explosive-based weapons can easily be manufactured, deployed and can have multiple effects. The need for protection of public environment and lives has given rise to environmental monitoring system that could quickly detect traces of explosives before causing havoc to lives. The selectivity and accuracy of sensors-based system is of paramount importance to have a system that is reliable in the presence of noise. This work tries to improve the selectivity and accuracy of explosive trace detection through the development of machine learning based model that utilizes the DTL approach. The model can work in a new environment with easy adaptation with limited dataset.

1.7.Definition of terms

- Conventional machine learning: this can be any machine learning approach other than deep learning that passes through the process of separate feature extraction process through which the machine learns
- Deep Learning (DL): it is a branch of machine learning model that uses backpropagation for pattern recognition without manual feature extraction.
- Edge devices: These are devices that serves as entry point into service provider core networks, they connect local area network to an external network for data accessibility everywhere.

- Explosive trace (ET): these are microscopic particles from explosive substances that can change the physical property of an environment.
- Sensor: device that response to variable input within the environment and give corresponding noticeable response that can be read by machines or human beings.
- Transfer Learning (ML): machine learning approach that uses the knowledge gained from a particular task to improve the performance of another related task

1.8.Organization of Thesis

This thesis has different chapters with each chapter contributing to the overall objectives of the work. It has the data acquisition part and the software development part to achieve the aim. Chapter one seeks to introduce the overall work, the aim and objectives, and scope of the research, while Chapter Two provides the detailed review of relevant literatures. Chapter Three covers the research framework, methodology, data used, the machine learning approach and the experimental setup. The experimental results and discussion are covered in Chapter Four. The conclusion is explained in Chapter Five with essential recommendations and possible further research.

CHAPTER TWO
LITERATURE REVIEW

2.1 Theoretical Framework

This chapter describes the basic concepts and theories related to this current research. Concepts discussed include Improvised Explosive Device (IED), explosive trace detection, wireless sensor networks, explosive trace detection, and machine learning approaches for explosive detection with focus on deep transfer network.

2.2 Improvised Explosive Device (IED)

Bombs that are manufactured at home or roadside using certain chemicals are IED whose manufacturing process does not follow the normal military conventional way of producing bombs. IEDs could be used by insurgents and terrorists for suicide mission and mass destructions of targeted areas. Since they are improvised, they can exist in divers' forms that could be like small pipe bomb or a form of sophisticated device that could cause massive destruction to lives and properties IEDs are usual hide in vehicle or carried by human beings, concealed in package; or place by the roadside (Gill et al., 2011). It contains an explosive substance that could be combine with other materials that could blast with the ability rippled destructive effect. These substances can be dynamite, gunpowder, and nitroglycerin, blasting caps, detonating fuses, black powder, and gunpowder. Some other substances could be combustible but not regarded as explosive because they do not emit ionizing, gasoline, oils, etc. are in that categories (Sapir & Giangrande, 2009). Some explosive chemical and compound are commonly available and can easily be accessible within most countries, civilian therefore easily manufacture IED illegally to cause civic unrest in the society(Wilkinson et al., 2007). IED can be used in any place, it can be dropped by the road side, brought into military barrack and area localized target.

Principally, IEDs is made up of an initiator, a detonator, an explosive charge, and a casing or collection of projectiles (such as ball bearings or nails) that produces lethal fragments upon detonation. In reality, it consists diver kinds of substance such as artillery or mortar rounds, aerial bombs, and some varieties of fertilizers, compound like TNT. It could as well contain radiological, chemical, or biological part to increase their lethal and psychological effect

(Mansoor, 2018). The effect of the IED mostly depend on the explosive used, those target at structure will have higher explosive to generate much more effect.

Explosives bombs can basically be classified into three main categories according to (AL-Mousawi & K. AL-Hassani, 2018). The first category is the military bomb which are said to be of a high standard in preparation and so need special intervention from governments. The manufacturing procedure is very complex and it requires high cost because of special devices involved that makes these types of bombs not readily available. The commercial or industrial bombs are the types of bombs that are used in the industry for process such as to detect metals and destruction of hard metallic substances. These categories are always developed in scientific laboratories. The third category is the improvised explosive devices that are mostly used by terrorists for unlawful attacks this is because it is easy to produce since the material for its production are readily available and as no special equipment is also required to manufacture it. This form of explosive can be of two categories with the first been an IEDs and the second category is the mobile type that like car bombs (AL-Mousawi & AL-Hassani, 2018).

Most commonly used approaches in explosive trace detection approach are sensor-based detection, Gas Chromatography (GC), Mass Spectrometry (MS), and Mobility Spectrometry (MS). The vapor and particulate emission are what the trace explosion detection approach utilizes for its detection. Different vapours are emitted from explosive particles such are used for research in explosive detection obtained from RDX, TNT and other explosive materials that include nitro aromatics, nitroaminies, nitroesters, acid salt, ammonium picrate, and organic peroxides. RDX is associated with the vapour nitroamines while the TNT is associated with nitro aromatics. Nitro amines for vapour trace explosive detection are associated with the explosives such as RDX: 1, 3, 5-triazacyclohexane, HMX: Octahydro- 1,3,5,7 –tetranitro-1,3,5,7 –tetrazocine and NQ: Nitro guanidine. Explosives associated with nitroaromatics include TNT: 2,4,6- Trinitrotoluene, TNB: 1,3,5- Trinitrobenzene, DNB: 1,3- Dinitrobenzene, 2, 4 DNT: 2,4 – Dinitrotoluene, 2, 6 –DNT: 2, 6 –dinitrotoluene, Tetryl: Methyl-2, 4, 6-trinitro-phenylnitramine, 2AmDNT: 2-amino-4, 6 – dinitrotoluene, 4AmDNT: 4-amino-2, 6 – dinitrotoluene, NT: Nitrotoluene (3 isomers), NB: Nitrobenzene and EGDN: Ethylene glycol dinitrate. Ammonium nitrate and urea nitrate are related to acid salt. Nitroesters are associated with NG: Nitroglycerin (glycerol trinitrate) and PETN: Pentaerythritol tetra nitrate. Picric acid relted exlsoive materials are AP/PA: Ammonium 2, 4, 6-trinitrophenoxide/2, 4, 6-

trinitrophenol while organic peroxides are TATP: Triacetone tripede and HMTD: Hexamethylene triperoxide diamine.

Most of the component used in manufacturing of IED products can be found with the description is shown in table 2.1. Most of these explosive components for the manufacture of IED can easily be found in medical stores and that makes the production easy.

Table 2.1: Common Explosive Component (AL-Mousawi & AL-Hassani, 2018).

SN	EXPLOSIVE SUBSTANCE	NATURE/WHERE TO BE FOUND	EXPLOSIVE IDENTITY
1.	Hydrogen peroxide	Can be found at chemist or pharmaceutical shop	IED
2.	Acetone	Can be found at polish remover, also as part plastic substance	IED
3.	Mercury	Can be found in dental stores in a form of toxic substance.	IED
4.	Ethyl alcohol	Can be called ethanol as well and can be found in medical shops.	IED/ military
5.	Methyl alcohol	Very flammable substance and can be used as antifreeze, can be called methanol and used as wood alcohol	IED
6.	Hexamine	This chemical substance removed from the white cool, white cool available on big stores.	IED
7.	Sodium acid	Available in medical store	IED
8.	Sodium nitrate	Known as Soda Niter, can be found at the agriculture stores	IED
9.	Ammonium nitrate IED	Available at agriculture stores	IED
10.	Potassium nitrate	Also known as nitrate, available at agriculture stores	IED

11.	Lead nitrate	Compound can be found at agriculture shops	IED
12.	Barium nitrate	Chemical compound can be found at agriculture stores	IED/ military
13.	Urea	Known as carbamide, available at the agriculture shops	IED
14.	Sodium carbonate	It is used to make papers and glasses, known as sal soda washing soda, and soda ash. Available at supermarkets	IED
15.	Sodium bicarbonate	A white soluble compound used in baking powder, known as baking soda, soda bicarbonate. Available at supermarkets	IED
16.	Ammonium hydroxide	Sometimes called ammonium water, can be found in supermarkets	IED / Military
17.	Potassium chlorate	Known as bleaching agent, chemical compound can be found at the supermarkets	IED
18.	Sulphur acid	Used as a vehicle's battery filler, known as battery acid	IED / Military
19.	Nitric acid	Available at the gold shops, known as aqua fortis	IED
20.	Aluminium powder	Available in painting store	IED
21.	Citric acid	A weak water-soluble acid. Can be found at the supermarkets	IED
22.	Acetic acid	The colourless pungent liquid widely used in the plastic manufacturing can be found at the supermarket.	IED
23.	Potassium permanganate	Used as a water cleaner, used in oxidising and bleaching agent, known as permanganate of potash,	IED
24.	Nitrobenzene	Oily high toxic water, used for screen cleaning, used to manufacturing aniline	IED
25.	Glycerin	Available in the medical store	IED / Military
26.	Petroleum Jelly	A semisolid mixture of hydrocarbons obtained	IED

		from petroleum, known in the market as (Vaseline), can be found in medical shops	
27.	Charcoal	This chemical element can be found at the leftover of wood burning	IED
28.	Hydrazine hydrate	Can be found on sponges	IED

These explosive chemicals can either be pure individual/single explosive or mixture of two or more chemical to produce the explosive. The single explosives can also be referred to every explosive compound that its unimolecular decomposition reaction may produce an explosion. Thy are pure compound that are consist of various atoms chemically bonded which can be of two categories of compound of either inorganic or organic as a result of the chemical compound (Zapata & García-Ruiz, 2021). Figure 2.1 show a comprehensive classification of explosive chemical. The pure individual explosive which is divided into organic and inorganic contain peroxide, nitro, organic azides, Halogen amino compound and other organic explosives, while the inorganics explosive contains the non-metal and metal explosives. The most common chemical found in this classification are TNT class, TATP class or the nitroguanidine (with one atom of carbon) while the TNT has seven atom of carbon and TATP has nine carbon atom. Others are the peroxides, Nitro explosive, Ammonium nitrate (NH_4NO_3), Chlorate-based explosives etc.(Zapata & García-Ruiz, 2021). Knowing these chemical constituents can the narrow the detections of the explosive trace to the response base on the sensor response to these chemicals. Majority of high explosives substance are described by this formula $C_a H_b N_c O_d$, it contains both oxidiser (O) and the fuel (C, H). Some of this substance can also have low sensitivity and that will demand high sensitivity sensor to be able to detect explosive trace using such (Jimenez & Navas, 2007). Most common explosive are nitrate base but the hydrogen peroxide has become popular because of terrorist. The approach appropriate for direct explosive traces detection in the form vapour that can detect explosive concentrate at below 1 ng/L.(Jimenez & Navas, 2007)

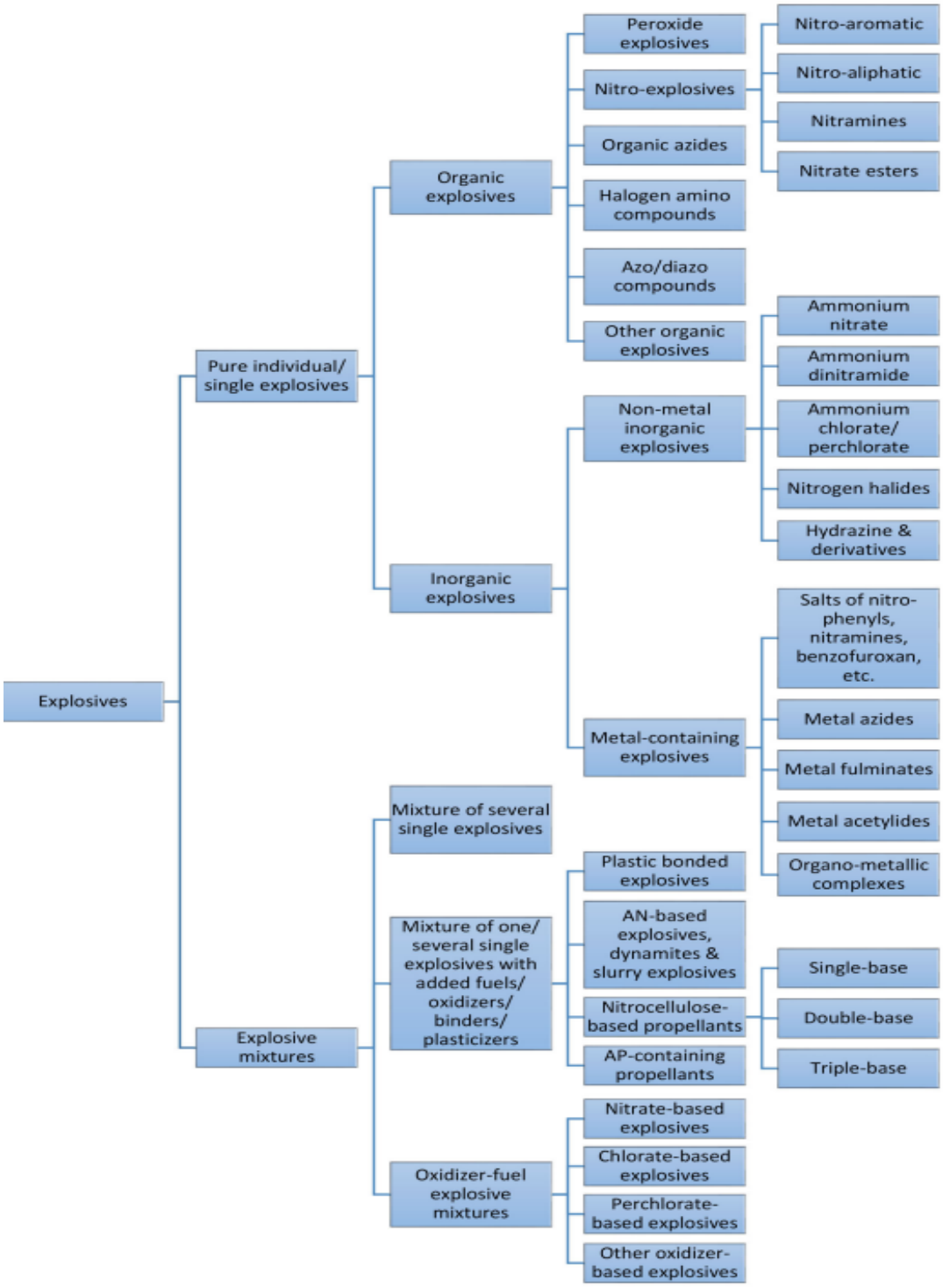


Figure 2. 1: Classification of Explosive Chemicals (Zapata H & García-Ruiz, 2021)

2.3 Explosive Trace Detection Methods

Generally, explosive detection approach can be categories in two ways, the trace detection method which focuses on vapor/particles that could lead to actual explosive and the bulk detection method which find actual explosive. Figure 2.2 shows the two methods of explosive

detection which are the bulk detection and the trace detection method. While our interest id on the trace detection approach, we shall give brief insight into the bulk detection approach.

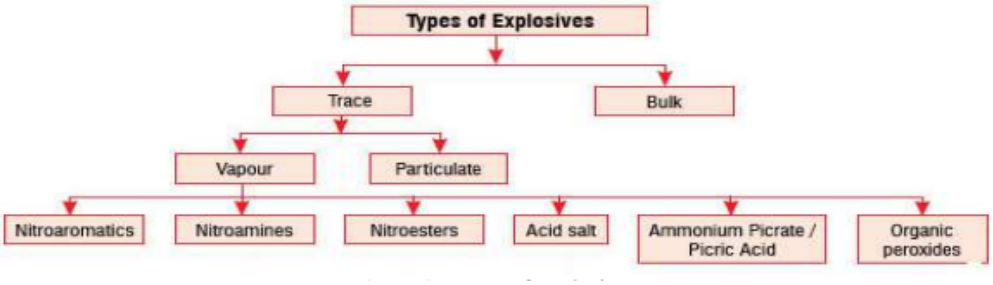


Figure 2.2: Explosive Detection Methods(Zafar et al., 2017)

2.3.1 Bulk detection method

The bulk explosive detection approach tends to detect explosives that are obvious to human that is big in size but sometimes maybe concealed, this approach tries to use high penetrating capacity system to clearly detect explosive presence. This methods of detection can either be imaging-based methods or nuclear-based methods(Kishore Kumar & Murali, 2016). The bulk detection approach apart from image target could also targets high nitrogen, oxygen content and high bulk density of the explosive substance (Marshall & Oxley, 2009).

Imaging approach such as various X-ray methods like single- energy X-ray, dual-energy X-ray, and computed tomography approach are employed for the bulk detection of the explosive. Most bombs have unique spatial features and specially shaped metal components like wires, detonators, and batteries. These components allow some level of discrimination from the background due to explosive dielectric constants for X-ray and microwave imaging approaches. The reflection, absorption, and scattering for various explosives in a set of spectral bands can be classified, and this information can be used as a data base for image analysis.

There are several imaging techniques that utilizes radiation with wavelengths from the range of radio waves to gamma rays(Kishore Kumar & Murali, 2016). Some method found in these techniques involve the use of X-Rays, Infrared, Terahertz and Microwaves. Another important method is the nuclear based approach that includes the use of thermal neutron analysis, pulsed fast neutron analysis, nuclear quadrupole resonance. Each method has its own advantages. In these detection methods screening of personnel in sensitive places, screening cars and items in the ship. Hidden bombs are searched to ensure protection of lives and infrastructure. One of

main concern in deploying these methods is health issue(The National Research Council, 2004).

2.3.3 The Trace Detection methods

Explosives can also be detected in a form of trace associated with the explosive. In this methods vapour/chemical emitted from explosive or explosive particles within the surrounding is used to detect the presence of explosive (Kumar et al., 2019). These explosives could exist either as a vapour or particulate form. If it appears as a vapour, it is found in the air but if it is in particulate is in a form of the residue of explosive material that adheres to surfaces of the object premise (Kishore Kumar & Murali, 2016).

In the trace detection approach, efforts are being made to track the chemical properties of the explosive substances and also its physical properties within the surrounding environment. Any of these chemicals has the ability to effect changes within the surroundings and this will greatly affect the properties of the environment (Al-mousawi & Al-hassani, 2017). The basic property used in explosive trace detection system is the chemical signatures of the explosive that easily alter many chemical constituents within the vicinity of its presence, this sudden in the surrounding of interest can be detected using chemical sensors like electronics nose. The other properties of consideration is the mechanical properties and physical properties *(Al-mousawi & Al-hassani, 2017).* The physical and mechanical properties are physical and all are related to how the explosive moves, the speed of movement and pressure, these properties are handled by mechanical sensor.

The vapour component is referred to as the gas molecules released from either solid or liquid explosive material. For proper detection of this trace some other information that are very important are the explosive concentration in the air known as vapour pressure, the frequency of explosives material in the environment, air flow in the environment , etc. The particulate are the tiny explosive substance that are like a form of leftover on the surface of object or human being that have made contact with the explosive through any means. The vapour sampling requires no contact while the particulate sampling requires direct contact to remove explosives material particles from a contaminated surface. This different form of explosive trace makes the detection system to have advantages and disadvantages in its approach (Thiesan et al., 2005). To overcome the setback, one of the best approaches is to consider the specific chemical

from the target compound in the material that is used to manufacture explosives rather generalized property. This help in reducing the probability of false alarm compared to bulk detection methods which focus on the typical property (Marshall & Oxley, 2009).

The three main categories of explosives are the Nitro aromatic explosives, Chlorate based explosives and the Peroxide based explosives. The traditional form of explosive detection has not been so effective because these techniques of detection of are large and terrorists can easily notice them and try to beat them(Zafar et al., 2017). Hence the need to use detection system that are not visible to human eyes and are economical to be set up in public places. Explosive detection evolution shows some natural beings like animal are efficient sensors for detecting explosive traces in an environment. Like the dogs could be trained on particular explosive material so well that anytime they smell the fragrancen of such material in the environment they can alarm their handler for the presence of explosive material. The limitation is that when dogs are tired of smelling they become ineffective. Honey bees are said to be most effective sensors used like the trained dogs but very difficult to harness and are not commercially available. Also considering the fact that some methods such as X-ray are visible and take more time to detect explosive (Zafar et al., 2017), the Wireless Sensor Network detection method has become a better method.

Automated means of detecting explosives is unavoidable in the present reality because of how frequent terrorist have started attacking and causing alarming destruction in sensitive's environment so it is necessary to have a workable intelligent system that could provide relevant information for needful actions against explosive based attacks (Kishore Kumar & Murali, 2016),. At a certain temperature, solids and liquids release vapour to the environment, the amount of vapour released can be used to determine the nature of that substance. The sample of this vapour are is collected without making contact with the surface of the material, sampling and analysis are air-borne. Since some explosive substance do not evaporate easily as a result material that depend on the vapour pressure hence, sampling strategies are very important due to the usually small amount of vapor- phase explosives material emitted from solid explosives material.

2.4 Machine Learning Approach in Explosive Trace Detection (ETD)

Machine learning (ML) is subset of Artificial Intelligent (AI) makes machine to receive data, analysis them and make decision without or with minimal human intervention, it's a trainable

assistant system adapting to individual user's objectives, the system can be trained to achieve what the user intended it to achieve (Díaz-Ramírez, 2021). Machine can also be trained to scan environment. As research in ML keeps progressing, we have seen recent development in intelligent systems that make systems to behave like human with capacity that enables systems to do the work of human beings (Shrestha et al., 2021)**.** This makes ML learning finds application is several fields like security. The development of a capacity-based systems that can solve advanced problem is generally referred to as artificial intelligence (AI), these systems used analytical approach algorithms to predictions, generate rules, give answers, recommendations, or similar outcomes in solving problems. This relieves humans of their burden and the risk of doing certain task by transferring their knowledge into a machine-accessible form and allow the development of an intelligent systems that will work efficiently (Díaz-Ramírez, 2021).

The fundamental concept of AI shall briefly be discussed for clarity and since Ai is not an entity, its relationships and differences with ML algorithm, Artificial Neural Networks and Deep neural networks shall be expressed. The Venn diagram in figure 2.3 shows the relationship between them. Generally, AI is the compound word that represent all technique that makes computers to learn how to be intelligent as human beings in reproducing whatever it learnt and making decision in solving complex tasks with minimal human intervention.

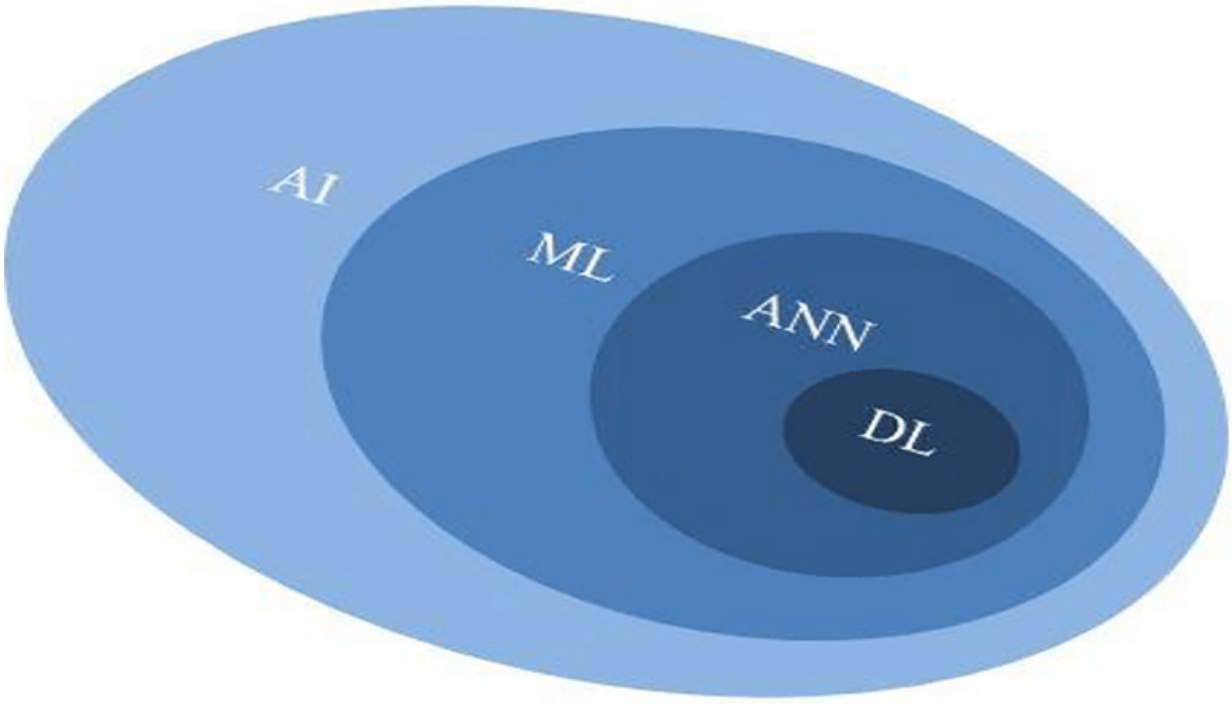


Figure 2.3: Relationship between AI, ML and DL (Aljojo et al., 2022)

AI research in the first stage is interested in hard-coded expressions that follow a set of rules that a computer can understand and carry out a logical decision. This is referred to as knowledge-based system and this is found to have limitations of not being able to handle complex task that ML approach has to solve (Díaz-Ramírez, 2021). ML is referred to computer program whose performance keeps improving as it keeps learning through experience with respect to assignment and certain performance measures (Jordan & Mitchell, 2015). It target how to automate assignment by using analytical approach to build performance cognitive tasks to detected object. It does that with the help of algorithms that keep learning from the task training data that makes the machine to have in-depth understanding of complex patterns even when fresh programming is not involve. When the system learns from pools of data that relate to classification, regression, and clustering, ML seems to be very reliable and perform that task with repeatable decision. ML model have gained success in several application area like image recognition, natural occurrence predations, natural language processing (NLP), etc. (Díaz-Ramírez, 2021). Generally, ML is divided into three types which are; supervised learning, unsupervised learning, and reinforcement learning. The supervised learning is majorly used in several applications that electronic markets.

Machine learning algorithm has some limitations such as inability to handle large volume of data (big data). Its approach focuses mainly on hand encrypted features which demand that the

system carefully learn those feature and extract them before taking decision based on these and these will take a lot of time. This model also has another limitation known as vanishing gradient and over fitting that tends to reduce the performances of the training models (Aljojo et al., 2022). These limitations are what leads to the emerging of Deep learning (DL), DL can handle complex data efficiently without experiencing the drawback of ML and this has made DL more acceptable than the traditional ML algorithm.

2.4.1 Traditional Machine Learning Concept

A group of methods and algorithms known as "traditional machine learning" were created prior to the development of deep learning. These algorithms are frequently used for many machine learning tasks, such as classification, regression, clustering, dimensionality reduction, and more. They are primarily based on mathematical optimization and statistical concepts. These are some of the fundamental ideas and techniques of conventional machine learning. Experience generates the matching algorithm model, and machine automated learning is truly the process that generates the algorithm model. Machine learning researches these learning algorithms (Zheng, 2023). The process of creating new things, reasoning with insufficient knowledge, digesting current big data trends, and replicating human thought processes are all included in the production of learning algorithms. Currently, supervised learning algorithms, unsupervised learning algorithms, and semisupervised learning algorithms comprise the majority of classical machine learning algorithms. Regression and classification algorithms are the two main categories of supervised learning algorithms. Using continuous functions to match input and output variables is known as regression. The matching of discrete categories and input variables is known as classification. Unsupervised learning implies that the final product is unknown beforehand. For instance, clustering allows us to extract a unique structure from the data. In unsupervised learning, there is either no label or just one label (Sarker, 2021) A learning strategy called semisupervised learning combines supervised and unsupervised learning. There are two types of data in machine learning: marked data and unmarked data. Learning may be made more accurate and efficient by using semisupervised learning.

A binary classification algorithm that supports both linear and nonlinear classification is called support vector machine (SVM). It is now commonly used in regression and classification and supports multivariate classification after evolution. It effectively resolves nonlinear, small sample, and high-dimensional issues and resolves the issues raised by conventional approaches. Experiments demonstrate that this approach excels in various domains and has grown to be an essential component of the machine learning community.

SVM is essentially a decision-making tool that classifies sample data; its true purpose is to solve. The classification problem is converted into a quadratic programming problem by first determining the maximum classification interval and then identifying the ideal classification hyperplane. The element problem is converted into a dual problem and subsequently into a convex quadratic programming problem by applying the Lagrangian optimization technique. In order to solve the optimization problem in this procedure, relaxation variables must be added if the sample points are linear and indivisible. The kernel function is utilized to solve the problem if the sample is nonlinear (Sarker, 2021).

With its strong classification performance, support vector machines (SVMs) have taken the machine learning world by storm since their invention.

Support vector machines are effective in high-dimensional spaces and can behave differently based on different mathematical functions. In high- or infinite-dimensional space, they construct a hyper-plane or set of hyper-planes. Intuitively, the hyper-plane, which has the greatest distance from the nearest training data points in any class, achieves a strong separation since, in general, the greater the margin, the lower the classifier's generalization error:(1) Gaussian radial basis kernel function; (2) Polynomial kernel function; and (3) Linear kernel function(4) The kernel function sigmoid

Another traditional machine learning approach is the K-nearest neighbors (KNN), it is referred to as a "lazy learning" method. It said to be "instance-based learning" or non-generalizing learning algorithm. It retains all instances corresponding to training data in n-dimensional space, rather than concentrating on building a generic internal model. KNN use similarity metrics, such as the Euclidean distance function, to classify new data points using existing data (Barupal & Fiehn, 2019). The k closest neighbors of each point vote with a simple majority to determine the classification. Accuracy is dependent on the quality of the data, but it is rather resilient to noisy training data. The most significant problem with KNN is determining the ideal number of neighbors to take into account. KNN is useful for regression as well as classification.

There are other traditional machine learning algorithm that can used for detection and classification of explosive trace data with appropriate parameter selection.

2.4.2 Deep Learning Approach in Explosive Trace Detection

The Deep Learning algorithms will record better perform whenever larger dataset is to be considered because it eradicates the challenge of vanishing gradient and overfitting which is a serious problem with traditional ML. It can bring out hidden information that are very relevance in large volume of dataset(Alom et al., 2019). Neural Networks (NN) is associated to ML, and that is where DL evolved from and since its emergence it has proved to be outstanding in almost all application domain. Deep Learning utilizes deep architectures or hierarchical learning approach, it is a subset of ML that became so pronounced from 2006 onward. Learning is a process that tries to estimate the system parameters so that the learned algorithm could carry out assigned task. The Artificial Neural Networks (ANN) uses the weight matrices as the parameter and it is made up of many layers in between the input and output layer that made it possible for non-linear data processing units with hierarchical architectures to be available for exploitation of feature learning and pattern classification (Schmidhuber, 2014).

DL has become so popular because of various successes it has recorded in complex data in object recognition detection and segmentation, image classification and localization, face and speech recognition and so on. In addition to these, DL is better intense of feature engineering, and feature extraction (Díaz-Ramírez, 2021). These advantages make DL the best model for explosive trace detection. Deep learning has become a strong analyzing model when dealing with high volumes of information that is generated through sensor network especially in an environment polluted with high level of noise and complex situation that make the conventional machine learning techniques difficult to be apply (Li et al., 2018). This problem can be easily solve by deploying Deep learning approach is seen as the best appropriate method since explosive traces has to be precisely detected in a complex and noise environment. The deep learning model with many layers can be scaled down to find sufficient features that can be applied to edge device.

2.4.3 Deep transfer learning for Explosive Trace Detection

DL tries to focus on reduction of training time of data especially when considering the cost implication of nonlinear data. Extensive training datasets is hard to retrieve in certain cases leads to the introduction of deep transfer learning (DTL). With DTL a pre-trained model for a certain assignment can be applied on a simple edge device like a cellphone that is limited in processing power and need reduced training time. Its developments has led to an intuitive and

high level of AI based systems because DTL sees learning as a continuous process (Iman & Arabnia, 2022).

In DTL, the model is first trained on one task, then the knowledge obtained from that model is used on another task or related task to reduce learning cost. A large amount of label data is necessary for accuracy in DL model and most time to get this dataset is very difficult and expensive, with the DTL higher accuracy can be gotten from small amount of trained dataset. There is also the need of reducing the processing power in ML models for it to work effective on edge devices like handsets, hence transfer learning is necessary.

To define a transfer learning using mathematical notation, let define what domain and task are. Say D is a domain that has two parts, made up of feature space X while $P(X)$ is marginal distribution. The Domain, $D = \{X , P(X)\}$. Where X is a symbol that shows the instance set and is

$$X = \{x|xi \in X , i = 1,...,n\}. \quad (2.1)$$

A task denoted as T with decision function f is made up of a space y, that is expressed as $T = \{Y, f\}$. f which is the decision function is to be learnt and generated from the dataset.
Certain machine learning algorithm usually gives the predicted conditional distributions of instances. This will yield,

$$f(X_j) = \{P(y_k|X_j)|y_k \in y, k = 1,...,|y|\}. \quad (2.2)$$

In essence, a domain is viewed and records the instances with or without the label data. For instance, say D_s is source domain that corresponds to source task, T_s that is always viewed through the instance-label pairs, that results in, $D_s = \{(x, y)|X_i \in X^s, y_i \in y^s, i = 1,...,n^s\}$; the target comprises of instances that is not labelled either any of few number of labeled instances.

The TL, when assigned certain observations that corresponds to $m^s \in N^+$ of the source domains and tasks, that implies $\{(D_{Si.} T_{Si})|i = 1,...,m^s\}$, and some observations relating to $m^T \in N^+$ target domains and tasks $\{(D_{Tj.} T_{tj})|j = 1,...,m^T\}$, what transfer learning does is that the knowledge gained from the source domains is been used to improve learned decision functions f^{Tj} $(j = 1,....,m^T)_{Si.} T_{Si})$ on the target domains. If $m^s = 1$, we will have a single-

source transfer learning scenario else it referred to as multisource transfer learning. m^T is the total sum of the TL assignment. Some research tend to make $m^T \geq 2$. The current transfer learning approach focuses more on scenarios where $m^S = m^T = 1$ (Zhuang et al., 2020)

DTL is not exactly same as semi-supervised learning, Multiview learning, i.e multitask leaning and another feature of the semi supervised learning is that the same dataset forms data source and target data. In this case the target data will not have labels, but Multiview learning approach differs because more than one datasets are utilized so as output of the task could be improved by the result obtained from another task. Like in video dataset, image data and audio data are separated. In the multitask transfer learning task are interconnected for the purpose of boosting each other while the transfer of knowledge happens the same time between all the tasks involved (Iman & Arabnia, 2022).

When we are considering the DTL, the interest is the target domain, and the knowledge needed for the target data is has been gotten from source data so, the is no need of running both the source and target data concurrently to obtain the necessary result (Iman & Arabnia, 2022). In the classification of DTLs groups based on label-setting three classes are considered; the transductive, inductive, and unsupervised approach. The transductive focuses on labeling the source data alone, inductive labels both source and the data of interest (target data) but in some cases none of the data is labeled it becomes unsupervised deep transfer (Zhuang et al., 2020). Another way to look at DTL approach is based on the aspect is been applied and this can be categorized into four which are: the instance-based, the feature based, parameter-based or network-based, and relational-based or adversarial-based. Instance-based transfer learning uses selected parts of instances sometimes all the instances in source data and apply different weighting approach on the target data to obtain result. In the case of Feature-based method instances or what is called features are mapped from both source and target data to form another homogenous dataset for result. The feature-based could be based on asymmetric approach of transfer learning or symmetric mode of transfer learning. While the Asymmetric approaches transform the source features to match the target ones, the symmetric approaches attempt to find a common latent feature space and then transform both the source and the target features into a new feature representation (Iman & Arabnia, 2022).

2.4.4 Edge-Computing Based Explosive Trace Detection

In edge-computing based system can be deployed to monitor the environment of interest against explosive trace, sensor network collect the ET information live and tries to communicate the information to edge computing server that will carry out expected data operation and analysis. This will result in reducing system energy consumption and network bottleneck (Fang et al., 2020). Edge computing is a modern technology that tends to minimizing the time the application will complete and the energy consumption of data transmission by distributing cloud resources closer to where data generation (Fang & Ma, 2021). The 'Edge' means to make computing device closer to the source of data. It is the distributed framework where data is processed as close to the originating data source possible and this tries to eliminate any form of delay in processing the output results. The processing of the data, storage and the networking get closer to the user.

Edge computing finds real application in WSN and this can be used to monitor the presence of ET in the environment. All the heavy ML model can be deployed in the cloud and the trained model can be deployed on the edge for real time prediction.

2.5 Sensor Network for Explosive Trace Monitoring

Sensor network compose of several sensors interconnected to monitor physical condition of an environment in real time, such condition could be temperature, pressure, pollutant sound, vibration or motion and explosive to produce sensory data that can be interpreted. The sensors can form an array or sometimes are connected through a wireless means called wireless Sensor network (WSN). The information gathered by the WSN is usually passed to the sink or base station which serves as an interface where human gets information from the network, which maybe through direct connection, satellite, internet, edge device or any type of wireless link (Fong, 2017).

A typical WSN contains several sensor nodes that can communicate with one another using radio signals. The sensor (radio) node consists of microcontroller, radio transceivers, power modules and external memory. The node can both serve as data originator and also data router. it receives what is sensed from the sensor and send it to the access point (sink node) (Suganya et al., 2019). The sink communicates with end user directly or through any wireless means. The sensors are used to capture the variable within the environment but convert this variable to electrical signal. The access point sent the data through the internet to the server where we have

the evaluation software. The challenge of deploying entity nodes in WSN are lack of resource associated with it, there is limitation of speed of response, storage space, and channel bandwidth and these are areas researchers have tried to address lately. When Global Positioning System (GPS) and certain model are used on the WSN location and positioning information can be obtained (Matin & Islam, 2018). Figure 2.4 show a typical WSN with multiple sensors, sink and user. Multiple user and sink could be in cooperated depending on the coverage area.

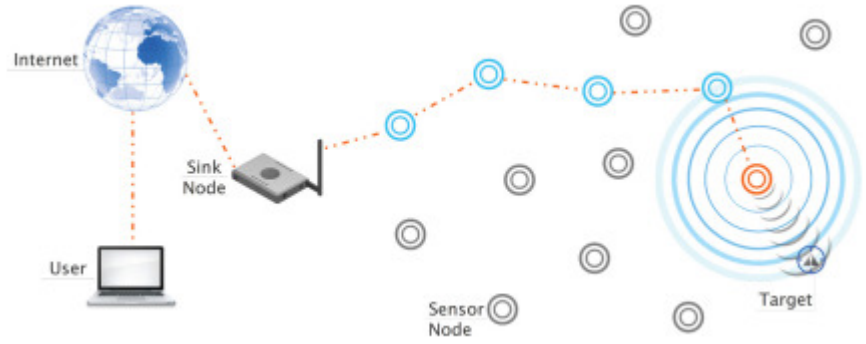


Figure 2.4: Typical WSN architecture (Matin & Islam, 2018)

When detection area to be considered is in a large public place, WSN must be deployed to cater such environment for adequate monitoring. With the help of WSN, a localized area can be monitored live using AI base system, data are collected, aggregated and forwarded to the server. When the defined characteristic is analyzed, prediction of explosive presence will be possible. Array of sensors forming WSN will be adequate to monitor the environment against the presence of explosive traces (Simi & Ramesh, 2011).

There are many types of sensors used for detecting system, Table 2.2, shows most sensor used for detection of explosive, all these sensors can be connected in different modes. The sensors that can easily be seen are those of thermal, photoreceptors, mechanical, chemical and physical sources. Table 2.2 is a descriptive table and characteristics of sensors according to (AL-Mousawi & K. AL-Hassani, 2018). The operation and description of the sensor will determine the quantity of the explosive it will be used to sense. We have different types of sensors ranging from chemical sensors, electrochemical sensors, and chromatography sensors and more, they are known based on the kind of sensing work they perform. Even though there are several types of explosives in existence that does not mean the same number of sensors are required. Explosives have some common features like geometry, material density, elemental

composition, and vapor emissions that could clearly classify explosive for proper choice of sensor. For geometry properties, image shape is used. When explosive density becomes the focus, it is believed that explosive is denser than other material. Another class is the vapour emission feature that senses vapour samples and analyze those (Kishore Kumar & Murali, 2016)

Table 2.2: Sensors and their Characteristic Properties (AL-Mousawi & AL-Hassani, 2018)

S/N	SENSOR TYPE	FEATURES OF SENSOR	ENVIRONMENTAL CHARACTERISTICS
1.	Piezoresistive pressure sensors	Membrane-in-cooperated in the sensor instrument that causes resistive changes leading to pressure power	Pressure power applied on sensor
2.	Capacitive pressure sensors	The effect of pressure on the surface of the sensor causes deflection and noticeable change in capacitance	
3.	Optical Pressure Sensor	The sensor response to laser light from optical fibre cable that produce noticeable change of colour in response to pressure changes.	
4.	Resistive and capacitive accelerometers	A Piezoresistive is connected to the sensor deflectable cantilever that response to any alteration in resistance to yield movement	Acceleration force presenting by velocity
5.	Piezoelectric accelerometers	This is piezoelectric types of sensor that yield charges when sensing materials is stranded	
6.	Electromechanical temperature sensors	Respond through the effect of temperature on the sensor material that yield electromechanical motion that could be interpreted in specific area.	Temperature and Heat

7.	Resistive temperature sensors	Effect of temperature on resistance result in same effect of sensor data variations due to the resistance effect.	
8.	Thermistors	This sensor type has resistor with deflectable material that is changed with slight changes in temperature	
9.	Resistive temperature detectors (RTDs) sensor covers larger temperature range	This sensor make use of metal oxide and resistive temperature detector to bring a noticeable changes in the environment, it consist of pure metal	
10.	Humidity sensors the substance volume	Calculate the ratio of water vapours	Vapours in substance
11.	Resistive Humidity Sensors	Calculate the resistive variations in a known medium	
12.	Chemical sensors	The sensor contains sensitive indicating transducer and membrane that could respond to chemical substance.	Chemical components and materials
13.	Interdigital transducer sensors	Very sensitive layer using dielectric between electrodes, the dielectric properties in sensitive layer are changed according to the substance interaction	
14.	Conductivity sensors as gases	Highly Sensitive layer that conduct flow of current and relate with chemical materials.	
15.	Optical chemical sensors	The sensor has a layer that easily deflect in the form of optical waveguide, that quickly respond when there is contact	

		between the substance chemical and chemical substance that is of interest	
16.	Piezoelectric chemical sensors	Gives rise to an electrical charge on crystalline when it is being stressed	
17.	Radiation sensors	Monitors level of radiation to detect beta and gamma in the material	Everything associated with radiation
18.	Geiger-Müller counter	Made up of conductors that response to the level of radiation.	
19.	Quartz fibre dosimeter	Calculate and report rate of radiation received over time from a device.	
20.	Film badge	Dosimeter calculates the rate of radiation level coming the material to the sensing element	
21.	Thermoluminescent Dosimeter Measuring Magnetic Sensor	Calculate rate of radiation from visible light in the material, it make use of magnetic field sensor.	Magnetic field
22.	Light and brightness sensor Gyroscope sensor	Consist of up of basic optical operation that have operate like structural and logical form	light level and colours degree rotation ratio circuits

2.5.1 Communication of Sensor Network for Explosives Trace Detection System

The means of data transmission between sensors in WSN does not need any form of cable connection but by use of wireless medium. One of the limitations of this means of communication is its limited range, so the sensors and nodes must be placed in locations that are not far from each other to effectively transmit data. This radio frequency range must be carefully identified depending on the type of application because each application has its specific frequency range. The applications could be industrial, health and scientific applications (ISM bands)(AL-Mousawi & AL-Hassani, 2018).

The area under consideration where the Sensor nodes are evenly placed is referred to as sensor field. The node gather data within a sensor field and transmit this data to where it can be processed within wireless sensor network system by utilizing multiple hop arrangement. Passing information from the sensors to the processor is done through the internet or via the satellite communication medium. The sink is the connecting link between sensor nodes and the processing unit. (AL-Mousawi & K. AL-Hassani, 2018). Whenever information is to be shared between the sensor fields, it will be done by the Base station, while information sharing among sensor nodes ad-hoc network comes into play. In the design of WSN large bandwidth is always been considered for adequate space for smooth data transfer.

2.5.2 Explosive Sensor communication models

The mode of communication for Wireless sensor nodes is through the radio units. A particular node is linked wirelessly with another node to both transmit and receive data between each other. Mathematical model can be used to express the connectivity and transmission between the two sensor nodes. One of common communication model employed is the disk connectivity model proposed by (Fong, 2017), he said communication between sensor nodes are only possible within a disk and that it occurs within the range of the radius, the radius of its communication range is called communication hub and that is the range two sensors could communicate. This focus of this model is to use geometric approach to analyze network connectivity which is quite simple in analysis but has limitation and not quite realistic because there is not clear boundary between the successful and unsuccessful communication.

Within a particular distance the attenuation of a wireless signal is a function of path loss and shadowing within that path referring to as path loss is express as to be the random fluctuations in signal strength. Through empirical measurements it was established that shadowing have proved to have zero-mean normally distributed random variable with standard deviation (SD) δ_{ϵ} (Fong, 2017). Environment varies in nature so most radio propagation models combines both analytical and empirical approaches determining path loss shadowing. The popular model is the radio propagation approach, by this log-normal shadowing path loss model which is given according to (Fong, 2017) as:

$$PL(d) = PL(d_r) + 10\gamma log_{10}\left(\frac{d}{d_r}\right) + \epsilon \quad (2.3)$$

PL is the path loss between transmitter and receiver, distance between transmitter-receiver is denoted as d, while d_f is a reference distance and γ is the signal decay rate or what we call path loss exponential, then ϵ is a zero-means Gaussian distributed random variable that has SD of δ_ϵ (dB) that expresses the shading effects.

At distance d, the output power of the transmitter less the $PL(d)$ is the received signal strength P_r. That is express as:

$$P_r(d) = P_t - PL(d) = P_t - PL(d_f) - 10\gamma log_{10}\left(\frac{d}{d_f}\right) - \epsilon \quad (2.4)$$

For a given value for which $\gamma = 2,\ \delta_\epsilon = 4,\ PL(d_f) = 15\,dB,\ d_f = 1,$ and for an output power

$P_t = 0\ dB$, the CC2420 IEEE 802.15.4, with 2.4GHz, the analytical propagation model is shown in Figure 2.4.

From equation (2.4), $P_r(d) \sim N(P_t - PL(d_f) - 10\gamma log_{10}\left(\frac{d}{d_f}\right)\delta_\epsilon,)$. Since $P_r(d)$ is a Gaussian, that information from sensor 1 get to sensor 2 as expected is expressed as a function of probability the two sensors , s_i *and* s_j located at distance d from each other is given as:

$$\rho[P_r(d) > SS_{min}] = Q\left(\frac{SS_{min} - \left(P_t - PL(P_f) - 10\gamma log_{10}\left(\frac{d}{d_f}\right)\right)}{\delta_\epsilon}\right) \quad (2.5)$$

Where $SS_{\epsilon mim}$ is referred as the minimum acceptable signal strength and Q is is the complementary cumulative distribution function of a standard Gaussian, so

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_{\bar{x}}^{\infty} e^{\frac{-t^2}{2}} dt \quad (2.6)$$

Figure 2.5 is the channel path through which the signal is transmitted, it is observed that the strength of the signal fades with distance. Figure 2.6 shows the formulated connectivity indicating how some area that supposed to receive connection are being discarded with receiving power less than *SSmin* while some other area receives power more than what is expected, i.e., beyond the connectivity range receive of *SSmin*. The main limitation is that there is no clear separation to determine the successful and unsuccessful communication among the sensors (Fong, 2017).

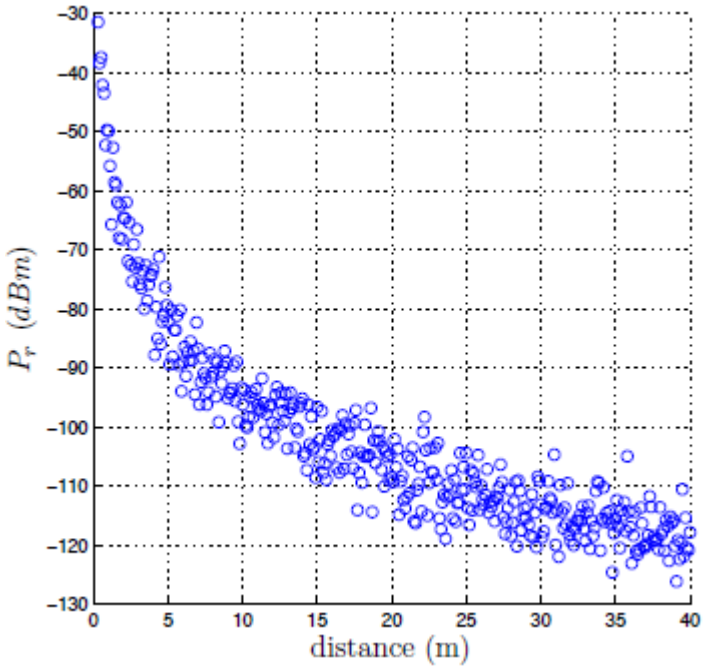


Figure 2.5: Channel Model, $\gamma = 2, \delta_e = 4$, and $P_t = 0\ dBm$ (Fong, 2017)

:

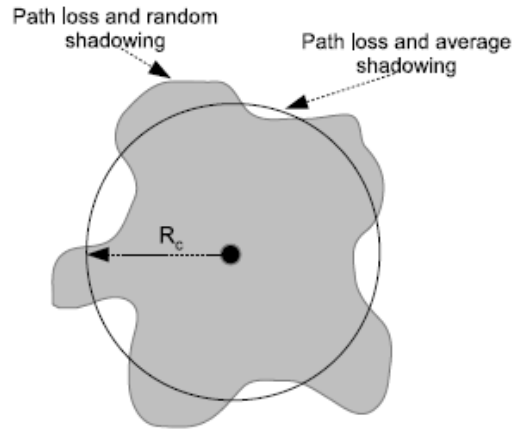


Figure 2.6: Connectivity model (Fong, 2017)

2.5.3 WSN for Explosive Trace Detection Deployment

The major challenge in deploying WSN is where to place the sensor for effective coverage within the area of interest to be surveyed and for this to be possible a careful optimization approach is used to achieve the design objective (Fong, 2017). The coverage objective is how to make sure the sensors are well arranged to maximize the area of interest and in achieving this the sensors must not be place too close or very far from each other. This will enable the sensing capacity to be fully maximized. For better performance on data acquisition in the localized area good deployment cannot be compromised. According to (Mao et al., 2019), two types of deployment approach are being employed which are deterministic and random deployment. In the deterministic approach the environment in question is familiar, the network functionality is rclaerly fixed and the sensor nodes are clearly placed in space. Mathematical model that are often transform into a linear programing problem or static optimization problem are being used to implement this. The hexagonal grids are usually used for nodes deployment when maximum network coverage and eff/ective connectivity is of great interest.

It was observed that deterministic model proved effective solution to deployment problem but it is oversimple and too perfect. However, when harsh environment is to be considered and also when large deployment is becoming difficult the random deployment will be the best option (Mao et al., 2019). One of the setbacks of the Random deployment method is the inability to guarantee full coverage, however, its cost effectiveness is a great advantage and when there is no strict coverage requirement it becomes a better choice. Some tines redundant nodes are introduced to achieve desired coverage.

Focusing on optimization object, the deployment of nodes is divided into three base deployment methods; coverage-based, network connectivity-based, and energy efficiency-based deployment. High result operation of the WSN is determined by how well the network is well represent in the target area. Node deployment of sensor network is greatly improved the performance and coverage of the sensors. (Mao et al., 2019).

2.6 Internet of things (IOT) Application in monitoring Explosive Trace

Environment of interest can be monitored and this process which is known as the process of capturing of values of data of interest in an outside environment. This approach can be used to acquire and grade the data even if it's a large volume of data (big data) using Internet of Things Technology (Cunin et al., 2018). When we connect and attach sensors to communicate with the

'various things', AI application can be used to enable these devices to share real-time data without human interventions. The IOT makes the society to be smart and flexible in adaptation and connect the digital technology with real-world. The different sensors that monitor environmental quality can be connected to IOT system to operate autonomously. Figure 2.7 shows IOT with different connection that can be anywhere, any environment and anytime. Whatever is sensed is transmitted and the output is viewed through web application or in some cases edged devices. The communication is through wireless means with appropriate communication protocol.

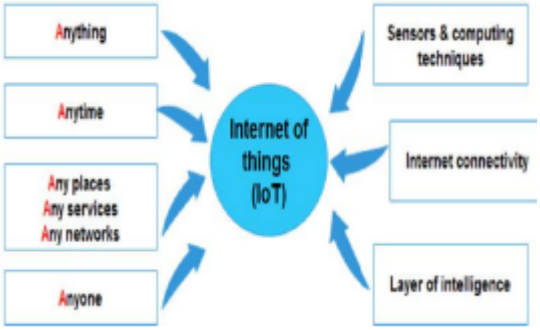


Figure 2.7: Concept of IOT in monitoring (Haji & Sallow, 2021)

Data collected by IoT environmental monitoring sensors in this case explosive traces within a wild area of interest. The environmental properties can be connected with a single cloud-based environmental system through the use of Wireless Sensor Network (WSN). When an IoT component fused with ML system can register, characterize, track, and analyze elements in a specific environment (Haji & Sallow, 2021)

2.6 Related Works

In this section, worked carried out such as biological means, analytical method, technological mean ranging from stand-alone sensors, sensor arrays, wireless sensor network and Artificial Intelligent (AI) approach in explosive trace detection will be reviewed.

2.6.1 Animal olfactory Systems

Chuen *et al.,* (2020) has shown that animal's methods have proved to exceed technological approaches in explosive trace detection especially for the fact that it can detect multiple traces of explosive concurrently and this is what sensor array network technology is finding difficult to achieve. Animal such as dogs, rats, pigs and honeybees are used to detect explosive traces.

Even though several animals are being used, dogs are commonly been used. In demonstrating the efficacy of dogs in explosive trace detection.

Command & Belvoir, (1978) shows a study that demonstrated the explosives trace detection by dog-handler teams which was carried out by Nolan and Gravitte, the teams were trained to detect landmines. The dogs recorded averaged detection of location accuracy of over 80% with several teams averaging 90% correct location. Further studies were carried out on this to examine the efficacy of detection teams, the training was improved and maintenance protocols developed by various agencies to validate the result scientifically. When the explosive was free from contamination and negative controls such as interfering samples, it resulted in improved accuracy of the detection. Another well-accepted work administered by the North American Police Work Dog Association that recorded about 91.6% accuracy on target odors was recorded in (Frost, 1990). The test was conducted on six different explosive odor classes over four of five different search areas. Although this publication was not peer-reviewed but was reviewed by panels of recognized experts before adoption. Sensitive and trained dogs were used by military during World war II to detect explosive and since then civilians began to used it for detection of drugs and explosive (Furton & Myers, 2001). Some of putative olfactory receptors from the dog detection have been cloned with subsequent characterization of some of the molecules. It has been shown that smell is the mechanism through which dog uses to detect explosive as dogs with defected sense of smell do not perform well in detection task. The Department of Defense program, which uses 500 explosive detection canines worldwide and has a proficiency requirement of at least 95% detection rate for the targets (known explosive odor standards) used and 5% or less nonproductive rate (alerts to distracter odors), is one specific example of how the reliability of explosive detection canines is repeatedly substantiated. (Thiesan et al., 2005). Among behavioral factors evaluated are type and duration of search, alertness of the team, responsiveness of the dog to the handler, and, the handler's skill in observing the behavior of the dog and interpreting those observations. Detection becomes more challenging since a living thing must be involved for accurate detection rather than relying solely on instrumental approaches. The U.S. Congress asked the Treasury Department to set standards for bomb-sniffing canines with the Bureau of Alcohol, Tobacco, and Firearms (ATF), suggesting the contentious standard of 100% accuracy on 60 tests. This move brought attention to the long-debated canine standards for bomb dogs. (Thiesan et al., 2005). Gazit et al., (2003), worked on implementation of a device that is used for operational research. The device aimed at assisting the handlers of sniffer dogs by the police to compare

the effectiveness of the dog in detecting explosive. The device was able to improve the efficiency of search in such operations. The device is to identify whether dogs utilized as a part of hunt are capable sniff or not. Those devices are incorporated with system that associates in remote recognition and investigation of explosives. While dog detection of explosive seems very good, adverse environmental conditions can easily affect it (i.e. high temperatures, long search times) and more prone to operator influence. The scientific knowledge acquired through the instrumental devices is generally more acceptable because it can be proved scientifically. For dog explosive detection calibration standards cannot be able to run and identify the specific explosive to make alert specific to the explosive type because the detector dog teams use sequential calibration (Thiesan et al., 2005). Dog training is quite costly because it takes a lot of time and effort to train them well. Animals are generally only useful for a few hours a day and have a tendency to become fatigued and distracted. This is a downside to their utilization. When they simultaneously detect explosive odors from multiple sources, they can become confused (Liu et al., 2019).

Under this biological approach used for explosive trace and some harmful chemicals detection Rodacy *et al.*, (2002). The honeybee's colony was used to cover a wider area that has different media such as land, water, air and plant, as they will be moving they came in contact with pollutants in the air,on plant and water that were in gaseous form, particulate or liquid form. These contamination are used to train the bees, in the process chemical such as 2,4,6-trinitrotoluene (TNT) have been used. Honeybees have been used to collect sample as well as locating contaminated areas and also to indicate anomalies in the area. In the experiment they setup sugar-water feeder closed to honeybee's colony and positioned explosive trace substances very close to it. The honeybees did not only got attracted to the sugar-water but also the explosive odor so that anywhere such substance is present in the future there will gather there thereby detecting explosive traces. In their work they achieve an accuracy of 98%. (Girotti et al., 2013) in their work Honeybees was used as biosensors since pesticides could affect their usage, they are then used to collect contaminant within the environment. Once there is changes in the environment, honeybee will behave differently and that can be a sign to predict explosive substance. (Bajić, 2014) tried to solve the challenge of locating the explosive trace through honeybees localization since honeybees has been accepted for explosive trace detection in a wild area. In their work they used methods such as
lidar, microwave dipole and detecting the third harmonic of the radar waves, and also using spectral features to detect the honeybees. Methods such as electro optical sensors, use of long

distance thermal camera combined with digital image processing have equally been deployed, in this case UAV was used. Although honeybees training is less and could cover more areas in detection of explosives trace like TNT, C4 and TATP explosives at parts-per-trillion levels but weather and night condition can easily affect the operation of the honeybees and may not also be deployed in areas where human beings are present (Chuen To et al., 2020)

Poling *et al.*, (2011) proposed the use of trained pouched rats for explosive like chemical detection. The training was carried out in a metal cage that has a small hole in the underneath and a pot was presented with a sample that has some small drop of about a 100ng per microliter of 2,4,6-trinitrotoluene (TNT). The person carrying out the training made a click sound and present mouthful of mashed bananas mixed that was mixed with crushed rat chow through a plastic syringe and the rat will smell through the nose for sometimes. When this positive training was ended the discriminative training was done where the trainer will serve the rats food along with the TNT sample by the hole and this will be done relatedly for several times until the rats only response when TNT sample is present with the food. They carried out the test using 34 rats with each rat covering 186,800 m2 and false alarms rate per $100m^2$ was 0.33 per 100 m2. While this has the advantage of portability and been less expensive, training takes much time and rats are not readily been available.

2.6.2 Analytical Approach for Explosive Trace Detection

Due to abilities of unique nature of the chemical constituents of explosive substances it is possible to analyze explosives contents as that was presented by the work of Wasilewski *et al.*, (2021). The method has aided in instruments design and development of the strategy aimed at detection of explosives trace with these systems. This method that is referred to as chromatographic includes the thin layer chromatography, gas chromatography (GC), high pressure liquid chromatography (LC), capillary electrophoresis, and ion chromatography; then spectroscopic or spectrometric methods such as infrared, ion mobility spectrometry (IMS), mass spectrometry (MS) are used. Shahraki et al., (2018) suggested using a negative ion mobility spectrometer in conjunction with an ionization source to detect explosives. In the investigation, explosive trace was detected using negative ion base thermal ionization operating in the air. In order to detect the mobility particle of typical explosion compounds like TNT and RDX in air, the ionization was enhanced by doping a chlorine chemical for the negative ion. It was determined that IMS is a highly regarded and frequently used technology in the majority of US airports for the detection of traces of nitro-organic explosives on carry-on luggage and

bags. One challenge is that since most explosives yields negative ions and most operated in the negative mode failed to detect trace on certain compounds e.g. TATP traces. To solve this problem of IMS not able to detect certain traces from some compound, Crocombe *et al.*, (2021) proposed the dual-tube IMS that could detect both negative and positive ions. Under IMS, sample vapors are often transformed into ions at atmospheric pressure, and the characteristics of those ions under mild electric fields are their gas phase nobilities. However, the vapor concentration dependency of the ion mobility spectrum and the seemingly erratic response caused by memory and humidity effects impeded the quick development of IMS and (Gary & Eiceman, 2006) has solved this problem earlier by developing an in-field analyzer that can best be represented by the handheld Chemical Agent Monitor. This development has made IMS to be found in most of the airport for screening against explosive substance. (Smith et al., 2020), developed flexible drift tube IMS system that is not expensive, the system that was constructed using a single printed circuit board was used to analysis common explosive substance such as RDX, TNT and PETNT and was found to have a detection limit of few nomogram and this make IMS device to be close to the substance meant to be screened. This had earlier been established by the work of (Mokalled et al., 2014), where qualitative analysis of a real explosion residue and explosive sample taken from a suspect was carried out and the explosive material and trace were identified successfully. It recorded a detection of explosives at Nano-gram levels and about six seconds response times, even with the little advantage of high speed in detection because it took only few seconds to detect explosive traces, its low selectivity was a serious drawback.

Evans-Nguyen *et al.,* (2021) proposed a fieldable Mass Spectrometry (MS) system used for security application. The system works based on membrane inlet systems and hybrid gas chromatography and the system recorded fast detection and also an improved selectivity. (Yinon, 2007) reported that the use of Mass Spectrometry (MS) for detection of explosive trace was based on the masses of the atoms and the molecule of the explosive substance. The mass to charge radio (m/e) is determine from the time and space of the charged substance in a force field. Since ions have difference m/e ratio they recorded different time of flight. A system was proposed to detect traces of explosive residues on aircraft boarding. The work focuses on how to detect traces of explosive residue on passengers that may had made contact with explosive substance and such substance would have been left on their body. An investigation of the quantities of explosive residues on previously used boarding cards was conducted. The residues are detected by the system prior to the passenger entering the aircraft and are transmitted by

touch to the boarding pass. A triple quadrupole mass spectrometer (MS/MS) was used to collect the generated vapors, which were then observed using selective reaction monitoring (SRM). Corona discharge is used to ionize the material. One of the produced ions is chosen to enter the collision cell and react with the nitrogen molecules there, producing a series of product ions. Precursor adduct ions are seen for RDX, PETN, and NG when an additive, such as dichloromethane, is added to the MS. Every hour, the system could process one thousand boarding passes. This result is based on a background investigation into the levels of explosive residues on two thousand boarding passengers. According to Yanon (2007), an explosive detection personnel portal is a walk-through system for quickly screening staff members for traces of explosives at locations like airports or federal buildings. This is an additional application of IMS in explosive detection. A mass spectrometer detector was used in the construction of the Syagen Guardian MS-ETD Portal (Grove et al., 2019). The following explosives were found: Tetryl, ammonium nitrate fuel oil (ANFO), triacetone triperoxide (TATP), hexamethylene triperoxide diamine (HMTD), RDX, HMX, PETN, EGDN, NG, and TNT.. Analysis time is less than 15s. MS recorded improvement in selectivity but its huge devices that are very expensive is required for large scale deployment of sensors in the wake of ever-increasing terror attacks prevailing in different part of the world was a limitation (Kishore et al., 2019).

Adegoke & Nic Daeid, (2021) proposed a method for explosive trace detection called "colorimetric optical Nano sensors for trace explosive detection using metal nanoparticles" The system is based on the work of Almog & Zitrin, (2009) that color reactions leads to the production of product that can be identify by its colour and this is a form of chemical reaction that is used to know the type of compound in used. So when you treat explosive compound with the right reagent can produce a unique colour that can be used to identify the constituent elements. Several system like Fluorescent and colorimetric sensors for selective detection of TNT and TNP explosives in aqueous medium proposed in Junaid *et al.*, (2022) have been developed based on this technology. The sensor based colorimetric system was able to respond to TNP and TNT substance very rapid. The fluorescence colorimetric methods is found to be one of the pronounce technique in detecting explosive trace. Fluorescence quenching methods remain the most popular technique. The major limitation of colorimetric method is the use of color reactions for the analysis of explosives that lies in their low specificity, some non-explosive chemical may produce the same colour and that is why colorimetric approach is combine with system to obtain the best result. It could only effectively work for specific

explosive or particular explosive compound designed to detect, for a wider range of field operation multiple colorimetry sensors have to be designed.

Remote detection of explosive trace using Raman Technology was presented by Hao *et al.,* (2022)**.** The technology was based on focusing of Laser Beam. They used two enhanced Raman spectroscopy methods to improve the low sensitivity observed in existing Raman Technology to detect explosive trace from Distance. In their method that used convex lens to converge the laser beam while collecting the Raman signal, the plasmonic spray was used to prevent Raman scattering along the surface. This enhanced approach achieved remote Raman detection up to thirty (30) meters different types of explosive with about 1 μg/cm2 of consecration. It was and improve version of Raman technology that was based on exciting a sample with a monochromatic light like laser, the explosive chemical composition radiate light at different frequency that can be differentiate from the from what exist in the environment. Raman spectroscopy is then used to collect the Roman spectra scattered light of the sample from a distance as a means of detecting substance that contain explosive trace (Gares *et al.*, 2016). This system involve the user of different types of laser that is hazardous to human safety especially the safety of the eyes. According to (Regis *et al.*, 2018) Another setback of the Raman technology is that fluorescent do interferes with its operation or when strongly absorbing substance is being used. Its operation fails on metals and it does not cover large area

2.6.3 Electronics Nose for Explosive Trace Detection

It was discovered that the usual electronic nose components are a chemical sensor array and an artificial neural network according to Liu et al., (2019). This array of sensors has unique properties that allow it to detect explosive traces of the target fragrance. Explosive traces are identified via an adaptive pattern recognition study of the signatures using techniques like artificial neural networks, and the pattern recognition process allows the identification of a particular explosive. According to Peveler *et al.* (2016), many electronics noses in array, such as a fluorophore array, were utilized for explosive chemical detection and discrimination. A quick reaction was achieved from a tiny amount of sample after array units were combined into a single multichannel platform. Using quantum dots as fluorescent probes, the multichannel platform detects and distinguishes between five explosives: TNT, DNT, Tetryl, PETN, and RDX. Another illustration was the colorimetric electronic nose, which was exhibited for the vapor phase detection and explosives classification. It was based on a handheld scanner and a

cross-reactive array (Askim et al., 2016). With a discriminating error rate of less than 1%, the array consisting of 40 colorimetric response sensors, 16 explosives including conventional explosives, characteristic explosive components, and homemade explosives was able to distinguish between 14 classes. Nonetheless, it is currently generally accepted that electronic noses are insufficient to identify the minute amounts of chemicals that dogs consume. Future developments will aim to expand the system's coverage and improve sensitivity and dependability. This improvement in the e-nose is what (Gradišek et al., 2019) used to utilize a 16-channel e-nose demonstrator that was based on micro-capacitive sensors with functionalized surfaces to measure the response of 30 different sensors to the vapours from 11 different substances, including the explosives 1,3,5-trinitro-1,3,5-triazinane (RDX), 1-methyl-2,4-dinitrobenzene (DNT) and 2-methyl-1,3,5-trinitrobenzene (TNT). In their work they developed a classification model through. Random Forest algorithm that was used to train set of signals, the varied parameters in their test were the concentration and flow of a selected single vapour. The model was able to recognizeand successfully classified the signal pattern of different sets of substances at an accuracy of 96%. Ot shows that the silane monolayers used in their sensors as receptor layers are can identify TNT and similar explosives from among other gaseous substances.

(Chowdhury et al., 2008) suggested a portable electronic nose system that uses five Metal Oxide Semiconductor (MOS) sensors that are available for purchase. A microcontroller is utilized to recognize patterns in the MOS sensors. Black tea scent is classified using the feed forward multilayer perceptron (FF-MLP) method in the IC (PIC18F4520). In order to determine the ideal architecture, weights, and biases of the neurons, the MLP is first trained using the backpropagation algorithm with the fingerprint from the sensor array and the corresponding tea tasters' mark in a PC. The samples were collected from various gardens in northeastern and eastern India. After training, the IC is programmed with the computed weights and biases of the neurons, enabling it to function as a portable device that provides the fragrance index for newly discovered tea samples without further processing. When compared to unidentified finished black tea samples, the results show that the ic-based electronic nose system performs on par with the PC-based electronic nose system.

Using 237 completed tea samples, the performance of the microcontroller-based electronic nose was assessed. Of these samples, 4000 patterns were evaluated for testing, while the remaining 60% were utilized to train the BP-MLP neural network model. In contrast to the PC-based electronic nose, which has an accuracy of 85.7%, the microcontroller-based electronic

nose obtained values of 83.6%. With a little quantity of tea samples, It was found that the accuracy of the FFMLP microcontroller-based electronic nose is somewhat less than that of the PC-based electronic nose. This could be because the portable version of the microcontroller uses an inbuilt 10-bit ADC, whereas the PC-based electronic nose uses a 16-bit A/D converter. Additional comparable work was completed by (Hasan et al., 2012), In order to detect spoilt meat kept in refrigerators, they created an electronic nose. The beef and fish samples were analyzed by an electronic nose, which then used a support vector machine (SVM) classifier to determine which meat was causing the bad stench. The experiment is run for a week in order to assess. The findings show that the SVM classifier performs well in generalization and allows for an accuracy rate of about 94.5% for both fish and meat. This indicates that SVM is a useful pattern classification method for identifying rotten meat using an electronic nose. With the addition of nano-enhanced sensors and changes in pattern recognition thanks to neural network technologies, electronic sensor work to mimic human nose sensing capability has improved and now can detect and identify minute amounts of explosive chemicals (Mokalled et al., 2014). The challenge of wider coverage has been confronting the Sensor designed for explosive trace Detection and to also have sensors that could detect multiple explosive the same time.

2.6.4 Sensor Network for Explosive Trace Detection

The sensor network technology tries to solve the problem of multiples sensing of explosive trace and also solve the problem of monitoring a localized environment against explosive trace. The sensor network is applicable to all the types of sensor used in detection explosive trace.

So *et al.,* (2009) proposed Laser-based atmospheric trace-gas sensors with great potential for long-term, real-time, maintenance free environmental monitoring in distributed Wireless Sensor Networks (WSN was proposed. A laser based chemical sensing technology with wide-area autonomous wireless sensor networking as the final target was developed. The prototype sensor measures atmospheric oxygen concentration in the form of a battery powered, handheld unit with power consumption <0.3W, sensitivity of 0.02% in 1 sec, weight of <0.4Kg without batteries, low cost, high specificity, and the robustness required for long term sensing applications. A gas plume localization and quantification using a prototype three-node sensor network was demonstrated. The technology is modular and can be used for different environmentally important molecules such as different environmentally important molecules such as CO_2 NO_x, and methane with exceptionally high specificity.

A reliable security threat warning system for public spaces like train stations, enabling security personnel to respond quickly to bomb threats was designed according to Simi & Ramesh, (2010). Using a multi-phase wireless sensor network, the technology offered a means of accurately and quickly detecting explosives in order to decrease, control, and alert people to impending terrorist action. The chemical makeup of explosives was determined using a number of wireless sensor nodes that were integrated with various kinds of sensors. The system dynamically collected data from the sensing nodes using several orthogonal strategies, aggregated the data, and forwarded it to the sink node for additional analysis. In order to verify the suspected items, a mobile node was subsequently added, improving the target tracking system and lowering the frequency of false alarms. In the work of (Simi & Ramesh, 2011) a multi-phase wireless sensor network design solution for monitoring was proposed. In order to lower the amount of false alarms, the system makes use of several wireless sensor nodes that are integrated with various sensor kinds and target tracking mechanisms. In order to respond quickly to bomb threats, this system offers an efficient warning mechanism for security risks in public areas. (Song et al., 2011) conducted research on the development and deployment of a wireless electronic nose (WEN) system that could identify and quantify the quantities of the flammable gases methane and (CH_4 / H_2). Two wireless sensor nodes in the system can function as either a slave or a master node. In slave mode, it consists of a wireless transceiver unit (WTU) that transmits the detection results to the master node connected to a computer, a digital signal processor (DSP) system that processes and samples sensor array data in real time, and a Fe_2O_3 gas sensing array for the detection of combustible gases. A Fe_2O_3 gas sensor type that is resistant to environmental effects is created that is insensitive to humidity. On a DSP, a threshold-based least square support vector regression (LS-SVR) estimator is used for concentration and classification calculations. The findings of the experiments verify that LS-SVR outperforms standard support vector regression (SVR) in terms of accuracy and convergence rate, outperforming artificial neural networks (ANNs). Gas mixture analysis is accomplished efficiently and in real time using the WEN system that was built. The system has limited application to be extended to other types of gases, particularly those associated with explosive trace.

(Rejeti et al., 2019) tried to establish the need to have a simple and effective network that can monitor an area against anti-social element such as explosive actions. They developed a detecting system that can detect explosives reliably and accurately. In their work a comprehensive framework that have all ingredients to detect explosives and integrated them

with a wireless sensor network (WSN). It was used to detect RDX and TNT explosives component. Explosive Detection Algorithm (EDA) was developed and proved to be effective. The simulation results shows great improvement over existing methods. Their work was not used to test other types of explosive component to show overall improvement.

In another development, explosive detection in border areas that handles threats from people and detect terrorist activities, they used PIR sensors for detecting person and metal detector was used for detecting explosives respectively, while a camera was used for continuous monitoring of the scenario at a remote station. They studied different technologies involved in the system. They include Bluetooth technology and infrared technology. They implemented a simulation study in Visual Basic using these three technologies (Minni & Siddharth, 2016).

Simi & Ramesh, (2011) designed a reliable security threat warning system for public spaces like train stations, enabling security personnel to respond quickly to bomb threats. By accurately and quickly detecting explosives, the system made use of a multi-phase wireless sensor network to provide a means of mitigating, controlling, and alerting people to impending terrorist action. The chemical makeup of explosives was determined using a number of wireless sensor nodes that were integrated with various kinds of sensors. The system constantly gathered data from the sensing nodes, aggregated it, and sent it to the sink node for additional analysis based on various orthogonal methodologies. In order to verify the suspicious items, a mobile node was added, improving the target tracking system and lowering the frequency of false alarms. Their system could not clearly discriminate against noise.

AL-Mousawi & AL-Hassani, (2018) to address the challenge of wider coverage of the sensor presented a work on wireless sensor network for explosive detection. Utilizing specialized sensors that are compatible with wireless sensor networks is necessary for explosive detection. The three primary axes of wireless sensor systems covered in this study are as follows: the first axis concerns the scalability of wireless sensors in explosives detection technologies. The connectivity and mobility of these networks and sensor are the second axes of the WSN explosives detection system. He discussed the need of using hyper sensor type that contains buddle of sensors for different simultaneous sensing. The challenge in WSN is the issue sensor security and latency, the WSN generally experience delay in transmitting information.

2.6.5 Artificial Intelligent in Explosive Trace Detection

The introduction of AI based technology in explosive trace detection is mainly to enhance the selectivity and sensitivity of the sensors and also try to solve the problem of latency in sensor network to achieve faster respond time. Different work has been done in this field.

Kapitanova *et al.,* (2010), suggested event detection, which is a key element in many applications involving wireless sensor networks (WSNs). We think that the frequently inaccurate sensor readings are too much for sharp values to manage. In their research, they showed that the accuracy of event detection is greatly increased when fuzzy values are used in place of crisp ones. They proved that a fuzzy logic method outperforms a few well-known classification algorithms in terms of detection precision. However, it was that using fuzzy logic has the drawback due to exponentially growing size of the rule-base. Sensor nodes have limited memory and storing large rule-bases could be a challenge. To address this issue, a number of techniques that help reduce the size of the rule-base by more than 70% while preserving the level of event detection accuracy was developed. Mølgaard *et al.,* (2017) offered a data-driven machine learning method for air sampling that uses colorimetric sensor technology to identify precursors of drugs and explosives. Utilized was the sensor technology developed within the framework of the CRIM-TRACK project. Currently, a fully functional portable prototype featuring automated data collection and disposable sensing chips has been created for air sampling. Large datasets of colorimetric data have been produced for several target analytes in laboratory and simulated real-world application scenarios thanks to the prototype's quick and easy sampling process. In order to reliably classify target analytes from confounders present in the air streams, many machine learning algorithms were utilized to leverage the very multi-variate data generated by the colorimetric chip. It was shown that relevant features and a high analyte detection rate can be obtained by combining a probabilistic classifier with a data-driven machine learning technique that uses dimensionality reduction. Moreover, the probabilistic machine learning methodology offers an automatic way to detect measures that are incorrect and may result in inaccurate predictions.

A series of studies concentrating on the amphetamine precursor phenylacetone and the improvised explosives pre-cursor hydrogen peroxide have been conducted to assess the durability of the colorimetric sensor. The investigation shows that, in real-world sampling

circumstances, the system can detect analytes in clean air and combined with naturally occurring chemicals. The technology being developed for CRIM-TRACK has the potential to be a useful tool for law enforcement applications such as bomb detection and drug trafficking control.

Deming et al., (2017) carried out a work on feasibility study where an artificial neural network was used to detect person-borne improvised explosive devices (IEDs) from images acquired from a radar array sensor, an infrared (IR) camera sensor, and a passive millimeter-wave camera sensor. The data set was obtained from the U.S. Department of Homeland Security (DHS) Science and Technology Directorate (S&T), and consists of hundreds of images of human subjects concealing various simulated IEDs, and clutter objects, beneath different types of clothing. The network used for detection is a hybrid, where feature extraction is performed using a multi-layer convolutional neural network, also known as a deep learning network, and final classification performed using a support vector machine (SVM). The performance of the combined network is scored using receiver operating curves for each IED type and sensor configuration. The results demonstrate (i) that deep learning is effective at extracting useful information from sensor imagery, and (ii) that performance is boosted significantly by combining complementary data from different sensor types. The focus of the work was not on trace and since images data where used the computational time is high. Their work only considered fabricated IEDs and not the possible properties. Al-mousawi & Al-mousawi, (2019) in their work represented a new direction in detecting magnetic explosives by the use of a wireless sensor network adapted with machine learning. The Improvised Explosives Devices (IED) consider a series threat due to the easy manufacturing. However, the scientific directions heading towards the use of information technology in the development of explosives detection systems. They focused on the type of explosives used is the magnetic explosives which are a type of IEDs that is used in targeting the vehicles. A Magnetic Explosives Detection System (MEDS) is a wireless sensor network system that uses a network of magnetic sensors to detect the magnetic field that emitted from magnetic effector and consider this magnetic field as a possible threat. The experiments of the system show its ability to detect the change in the magnetic field caused by the magnet stacked under the vehicle. The main use of the neural network algorithm in this paper is to determine the highest reading among a series of readings to determine where the threat exact position. Excellent results produced by the neural network algorithm in the MEDS to enable the system from learn and identify the required data type.

Fisher *et al.,* (2020) proposed machine learning approach to improving Trace Explosive Selectivity which they applied to Nitrate-Based Explosives. In the work, machine learning methods were utilized to examine the extent of improvement in IMS selectivity for detection of nitrate-based explosives. The work considered five classes: ammonium nitrate (AN), an ~ 95:5 mixture of AN and fuel oil (ANFO), urea nitrate (UN), nitrate due to environmental pollution, and samples that did not contain any explosive (blanks). The preliminary results clearly show that the incorporation of machine learning methods can lead to a significant improvement in IMS selectivity. (Liu et al., 2019). The effort to enhance the selectivity and reliability several work have been done such as (López et al., 2017) that used Principal component Analysis (PCA) on metal oxide sensor (MOX) array and that was able to identify explosive samples and discriminated between them and other substances like Ethanol and Vinegar, a k-Nearest Neighborhood algorithm was used with k equal to 3. A Leave-one-out cross-validation strategy was established to estimate the classification rate of the final model which they recorded as 86%. (Wang et al., 2005) had previously used a novel intelligent technique based on support vector machine (SVM) classification for electronic nasal signal detection. SVM functions under the tenet of minimizing structural risk, which ensures improved generalization capacity. After demonstrating the SVM's fundamental idea, the gas classifications were recognized using the SVM as a classifier. The method can overcome the drawbacks of artificial neural networks by classifying complex patterns, achieving a higher recognition rate at a reasonably small size of training sample set. There has been a presentation and discussion of the tests conducted to identify three distinct gases: acetone, gasoline, and ethanol. The findings show that the SVM classifier performs well in generalization and raises the tested samples' average recognition rate to 88.33%. This indicates that the suggested approach for electronic nose signal recognition is successful. One of the area of electronic nose sensor to improve is the area of selectivity and this to to enhance the accuracy of detection of the analytes.

In further development, the application of a convolutional neural network (CNN) to facilitate IED detection was proposed by Colreavy-donnelly et al.,(2020). An autonomous sensor array was utilized in a related research to find the devices in areas that were too dangerous for a person to survey. CNN and its training approach are appropriate for using the sensor system in this work. In real time, this convolutional neural network can detect and discriminate between natural features of the surrounding undergrowth and a potential IED. In well-lit environments, the CNN was able to identify the IEDs with 98.7% accuracy because to the training process.

The suggested CNN performs better than its rivals, including the deterministic approach, when the results are compared to those of other convolutional neural networks and a deterministic algorithm. The limitation of his work is that the environment must be well illuminated before high accuracy could be recorded, what happen if the attack is to take place in a dark environment.

According Fisher *et al.,* (2020), the preferred technique for finding traces of explosives in most airport and border crossing environments is ion mobility spectrometry (IMS). The IMS detection limits are low enough to meet security standards for the majority of explosives. Nonetheless, the selectivity is insufficient for certain explosive families. One such category of explosives is nitrate-based explosives, where it can be difficult to distinguish between different nitrate hazards and ambient nitrates. Machine learning techniques were applied to investigate the degree of enhancement in IMS selectivity for nitrate-based explosives detection, using a limited database. This exploratory investigation looked at five kinds of nitrate: urea nitrate (UN), nitrate from environmental contamination, ammonium nitrate (AN), a ˆ95:5 mixture of AN and fuel oil (ANFO), and samples free of explosives (blanks). The initial findings unequivocally demonstrate that applying machine learning techniques can significantly increase IMS selectivity. Zapata & García-Ruiz, (2021) conducted thorough analyses of a few basic ideas regarding explosives and the two widely used categories of them according to either their application or their velocity of detonation. They claim that while the current classifications are very helpful in the legal and military spheres, they are of no use in figuring out the chemical makeup of explosives. The classification of explosives according to their chemical makeup was the main topic of their review. This classification succeeded in creating a distinct general classification by combining the chemical classifications of explosives present in literature. Explosive was classified into single explosive and mixture explosive; the single explosive was further classified into organic and inorganic explosive. The work provides adequate knowledge of the chemical composition of explosives but did not indicate the appropriate sensor to indicate the presence of these chemical composition.

Wongwattanaporn, (2021) proposed a way of finding a suitable classification technique to be implemented in an electronic nose so as to imitate the sniffer dogs in detecting the explosive chemical substances. In the work, eight different classification techniques, which are Logistic Regression, Support Vector Machine (SVM), Decision Tree, Random Forest (RF), Adaptive Boosting, K-Nearest Neighbors, Gaussian Naive Bayes, and Multilayer Perceptron in both

binary and multi-class gas sensor array open-source datasets where compare in terms of accuracy of detection. The experimental results show that RF and SVM models perform better with average score of 99.66 and 98.93, respectively. Much data where needed to carry out the training, in real life scenarios adequate data may not be available for training of the model which may reduce the accuracy of the model, he did not equally focus of detecting the explosive trace within an area.

Djedidi et al., (2021) suggested an innovative method for detecting the presence of one of the three harmful gases—CO, NO2, or O3—either alone or in mixes, relying on a single physical sensor and data-driven algorithms. In the hardware portion of the project, a single Metal Oxide (MOX) sensor was connected to two heaters. A supervised machine learning model was implemented in the software portion. The sensor changes its electric signals in response to the various gases and their mixtures that it is subjected to. The core dataset for the discrimination consists of these raw signals and the heater readings. The raw dataset is enhanced by computing multiple time-domain characteristics for every measurement in order to improve the classification results even more. Following a ranking of the characteristics, the features that best address the categorization problem are chosen. Following data preprocessing, a multi-Support Vector Machine model is trained and validated using the features that were chosen. The system was able to detect and classify the various gases with high accuracy, but with the use of multi-Support Vector model computation time will be high and when you don't have much data, it will affect the accuracy of the system.

2.7 Summary of Literature and Research Gap

The review of similar works on Explosive Trace Detection (ETD) can be classified under the categories and the gap as identified from literature summarized as followed:

The use of Animal in Explosive Trace Detection: It was established that animal such as dogs, rats and bees is one of the best method for detecting explosive trace and they are currently still been used **(**Chuen *et al.,* 2020**)**. They could detect multiple analytes the same time and Bees particularly can be used to monitor large area against explosive trace. Generally the use of animals has drawback due to the tendency of the animal like dogs and rats getting distracted and tired and can only be used effectively for a few hours a day. Sometimes they get confused in case they smell explosive from several sources simultaneously. The use of animals can be restricted in areas where human beings are present. It is equally very expensive to train animals

for explosive trace detection and to train them for such it require long period of time (Kishore *et al.*, 2019). There is therefore need to develop a system that can monitor an environment of interest irrespective of the terrine of the environment and such system can work independently without constant human intervention. Since the system will be setup once it will reduce cost compare with the use of animal to detect explosive traces.

Explosive Trace Detection based on analytical instruments: in this approach a method called chromatography is used, gas chromatography (GC), high pressure liquid chromatography (LC), and ion chromatography (Wasilewski *et al.*, 2021). Pronounced method involves spectroscopic or spectrometric methods such as infrared, ion mobility spectrometry (IMS), mass spectrometry (MS), and Colorimetric and Raman technology. This approach uses two sets of anionic and cationic analytical methods after conversion of the chemicals to respective ions to allow identification and confirmation of the presence of inorganic explosive residue. One of the major drawback of this method is that these devices are bulky and highly expensive to deploy to tackle the challenges of increasing terrorism. Inability of the technologies to monitor large area is also a limitation. Since sensor can be so tinny, the sensor network base approach can be hidden and thereby become invisible to people carry explosives, into the secure areas. This make the proposed method viable to be deployed in an area without terrorists knowing that such detector system is present in the environment.

Electronics Nose in Explosive trace detection: Electronic nose which is a technological device designed to mimics animal in sensing explosive substance explores the biological olfactory function. Its ability to distinguish complex volatiles substances, makes it unique to the principle of olfactory system. The Electronics nose has the sensing part and the artificial neural network that makes the system achieve better results (Liu *et al.,* 2019). The limitation is that they may not be able to detect multiple analytes the same time and if they are to achieve that there must be in array. The array of sensor can comprise several sensors types and thereby be able to detect multiple types of explosive trace. This make the system highly reliable.

The Sensor Network: For the Electronics nose to have wider applications and coverage the sensor array network and wireless sensor networks are been introduced. Sensor array were built for detection of multiple constituents, while WSN where used for wide area cover but one major challenge here is the problem of latency and sensitivity of the sensors (Al-mousawi & Al-

mousawi, 2019). Solving the challenge of sensitity has made machine learning model a better approach in designing a sensity system that could detect explosive traces with high accuracy.

Machine Learning in Explosive Trace Detection: For accuracy of detection and high sensitivity of Sensor network that detect explosive trace, different machine Learning Algorithm are been developed in Explosive Trace Detection. Convectional Machine Learning model such as Support Vector Machine, KNN and CNN where used and to achieve high sensitivity and accuracy the Deep Learning Model such as Convolutional Neural Network was proposed and achieve better results compare to other methods (Wongwattanaporn, 2021). Deep learning was also introduced to improve selectivity, it achieved that with large dataset that could not solve the problem of latency. The limitation of the traditional machine learning model is that the system was more accurate on what it has been trained for and much data were also need

The proposed method is Deep Transfer Learning for Explosive Trace Detection (DTLETD) is effective in solving the problem of explosive trace detection from limited data. Explosive trace data are very limited because of the nature of restrictions in acquiring the data and also the cost. There is need to develop a system based on DTLETD that can work on edge device that will be light in size with high accuracy of detection of explosive trace in the presence of other chemicals. The development of lighter weighted model that could be used on edge device while solving accuracy problem is one of the main problem DTLETD tries to solve.

CHAPTER THREE
METHODOLOGY

3.1 Problem Formulation

The security of a localize place against the high rise of explosive attack has become inevitable and the electronics sensor network play a significant role in being used to detect traces of explosive within an area to be secured. For accurate, precise and timely explosive trace detection system, machine leaning approach has been used for sensor base detection system but using the existing machine learning approach could only detect traces that the models are familiar with. The issue of lack of explosive data and longer time of training using deep learning is also a problem desiring solution. To solve the problems, there is need to develop Deep Transfer Learning for Explosive Trace (DTLET) model based on CNN model. The proposed method will be developed based on the CNN model and is expected to be accurate and light weight for deployment on edge devices. The system is expected to be fast in detection with little dataset.

3.2 Proposed Framework for Explosive Trace Detection

The proposed framework for area based explosive trace detection system consist of a CNN based model known as GasNet that will be fine-tuned then reconstructed into a new model, this model was used to train explosive trace data from scratch as indicated in Figure 3.1. The knowledge gained was used to test and validated explosive trace data from the sensors array. The sensors are placed within the environment of interest to form array of sensor network which can be in a form of Wireless Sensor Network. These sensors respond to explosive trace and their response are converted to electrical signal that will be converted to digital signal that form the new input to the Deep Learning transfer model for explosive trace detection. Figure 3.1 shows the complete framework for area based explosive trace detection consisting of sensor network, signal conditioning, signal conversion process and the proposed model of DTLETD. The adopted base model is GasNet (Pai *et al.*,2018) which

will be fine-tuned with appropriate layer adjustment and then the learned knowledge will be transferred to predict explosive trace. The validation data was generation from the sensor in the implementation model of the area-based explosive trace detection system. This set up is meant to generate data for the developed model for validation. The result of the prediction will be used to notify the appropriate authority for corresponding actions.

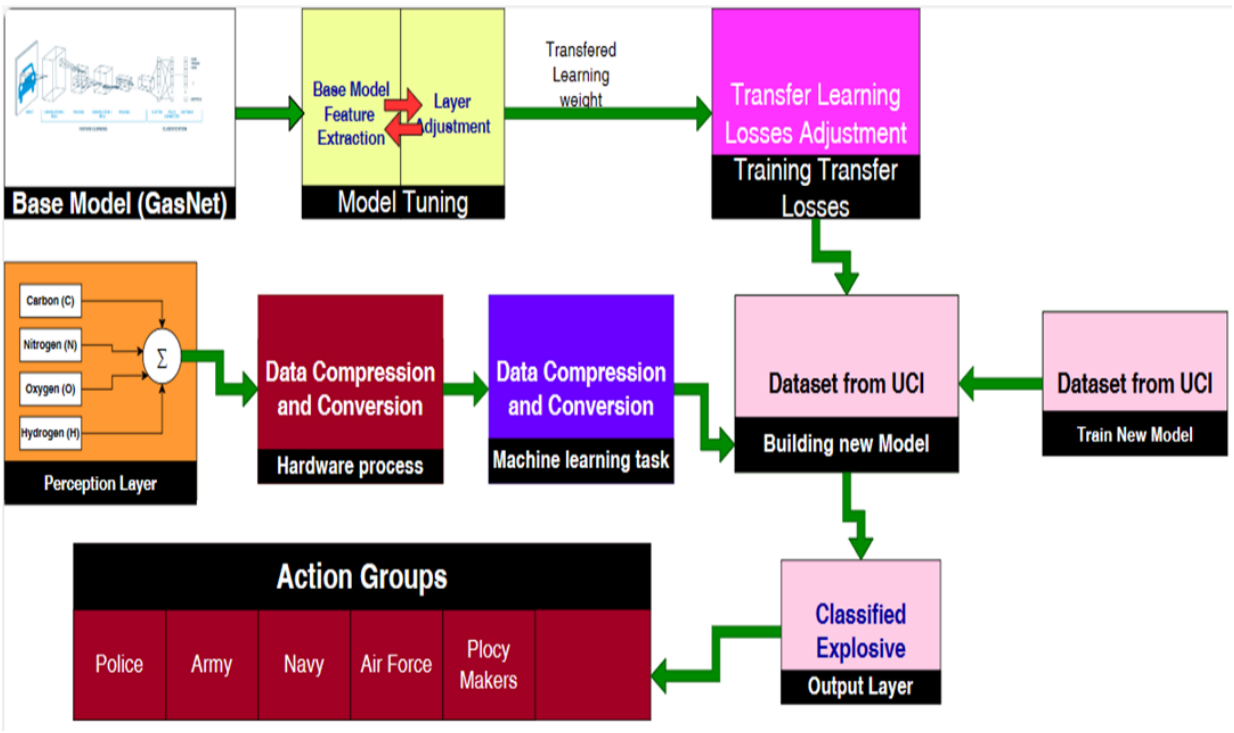


Figure 3.1: Conceptual Framework of DTLETD

3.3 Research Process and activities

The explosive trace Detection system shall involve development of an accurate classifier for the detection of substances containing explosive is achieve through series of operational activities. Figure 3.2 shows the training model activity program which involve receiving a Deep convolution neural neatwork (GasNet) model that is used to train explosive trace dataset, the model is adjusted and fine-tuned to form a new base model. The training and testing performance results was obtained for both training dataset and testing dataset and will be

presented in Chapter four. Figure 3.2 shows the Activity diagram for training the model of how input data is received to the model, layers freeze and reconstructed to obtain the new model.

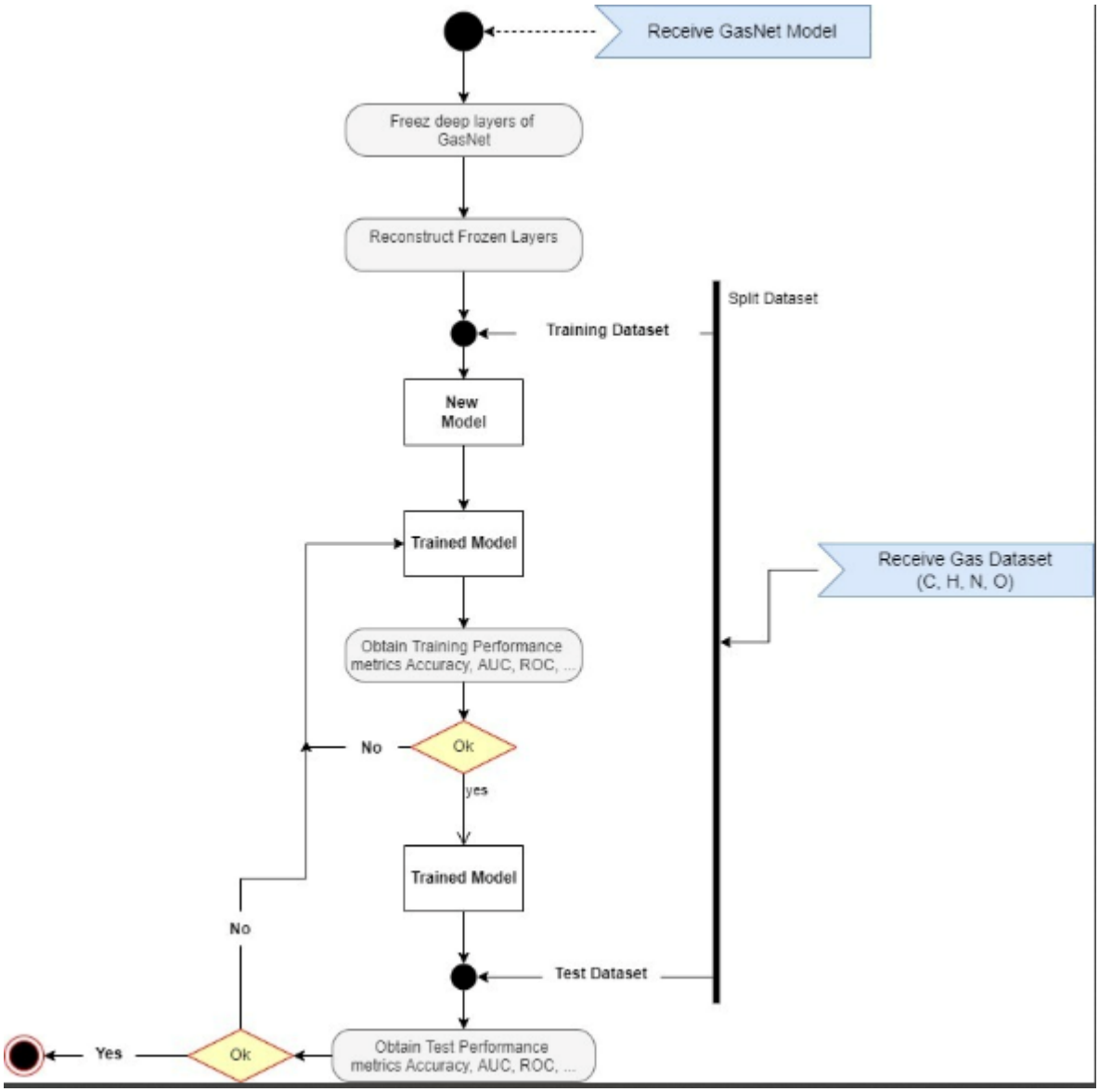


Figure 3.2: Activity Diagram for Training the model

The development activity diagram for the transfer learning model is shown in Figure 3.3 where the input data is from the sensor array network. The sensor data is Preprocessed and used on the developed model based on knowledge gained from Figure 3.2 to classify explosive trace. The performance of the system is evaluated for corresponding results.

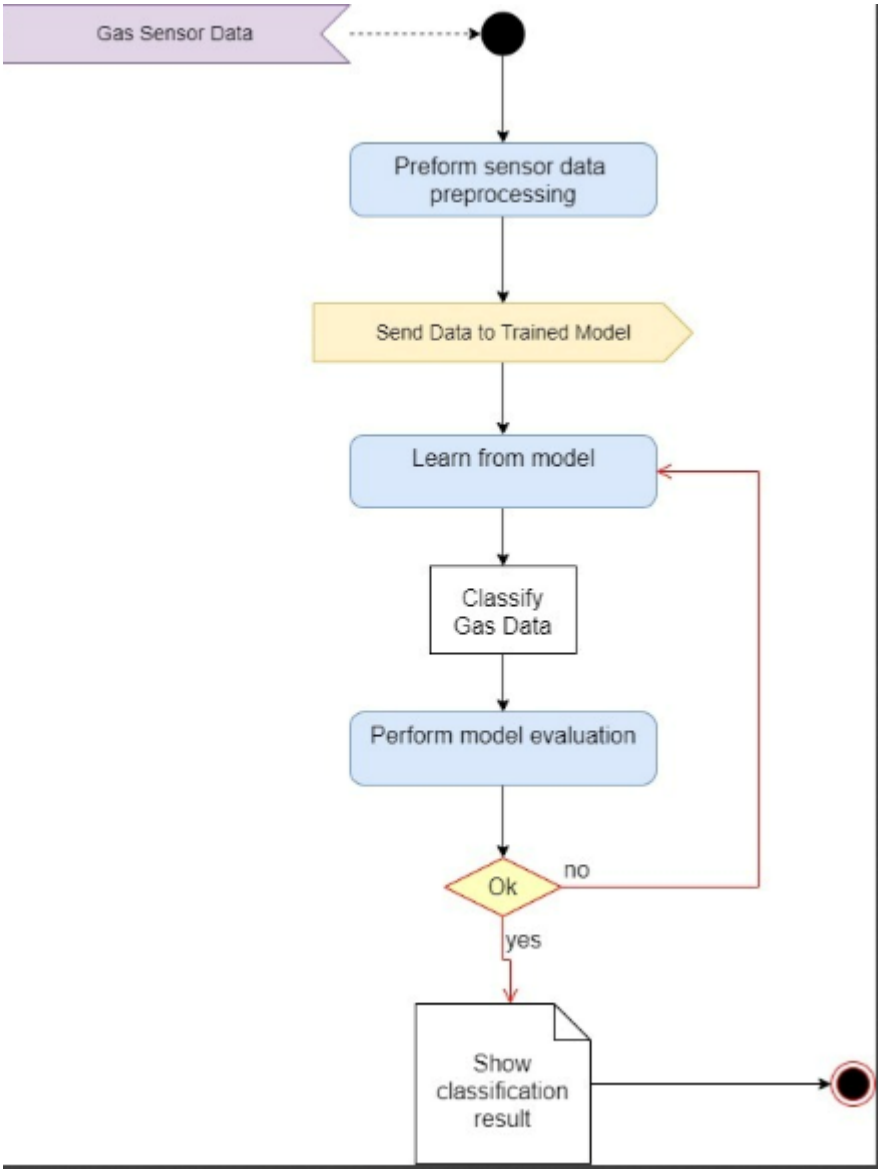


Figure 3.3: Development Activity Diagram for deep transfer learning

3.4 Deep Transfer Learning for Explosive Trace Detection Model

In this work, we utilized the possibility of Deep Learning (DL) model to accurately detect explosive traces results with limited explosive trace data set collection which is important for future experiment design that could detect explosive trace with limited explosive trace data on edge devices. The system is to detect the explosive trace very fast with high accuracy. The Deep transfer learning model is developed to be accurate, fast and light-weight classification that can be deployed in sensor network in order to identify explosive substances within an environment. Since the deep learning model requires considerable large volume of explosive data for better performance. Explosive trace data is very scarce because of the restriction in the

manufacturing of explosive and the precursors of same. It will be very necessary to use readily available data with similar characteristic to first train the model before implementing it on explosive trace data. The conceptual model of Deep transfer learning is shown in Figure 3.4. Available online dataset from gaseous pollutant will be used as source data, while target data will be the explosive trace dataset collected from Sensor Network.

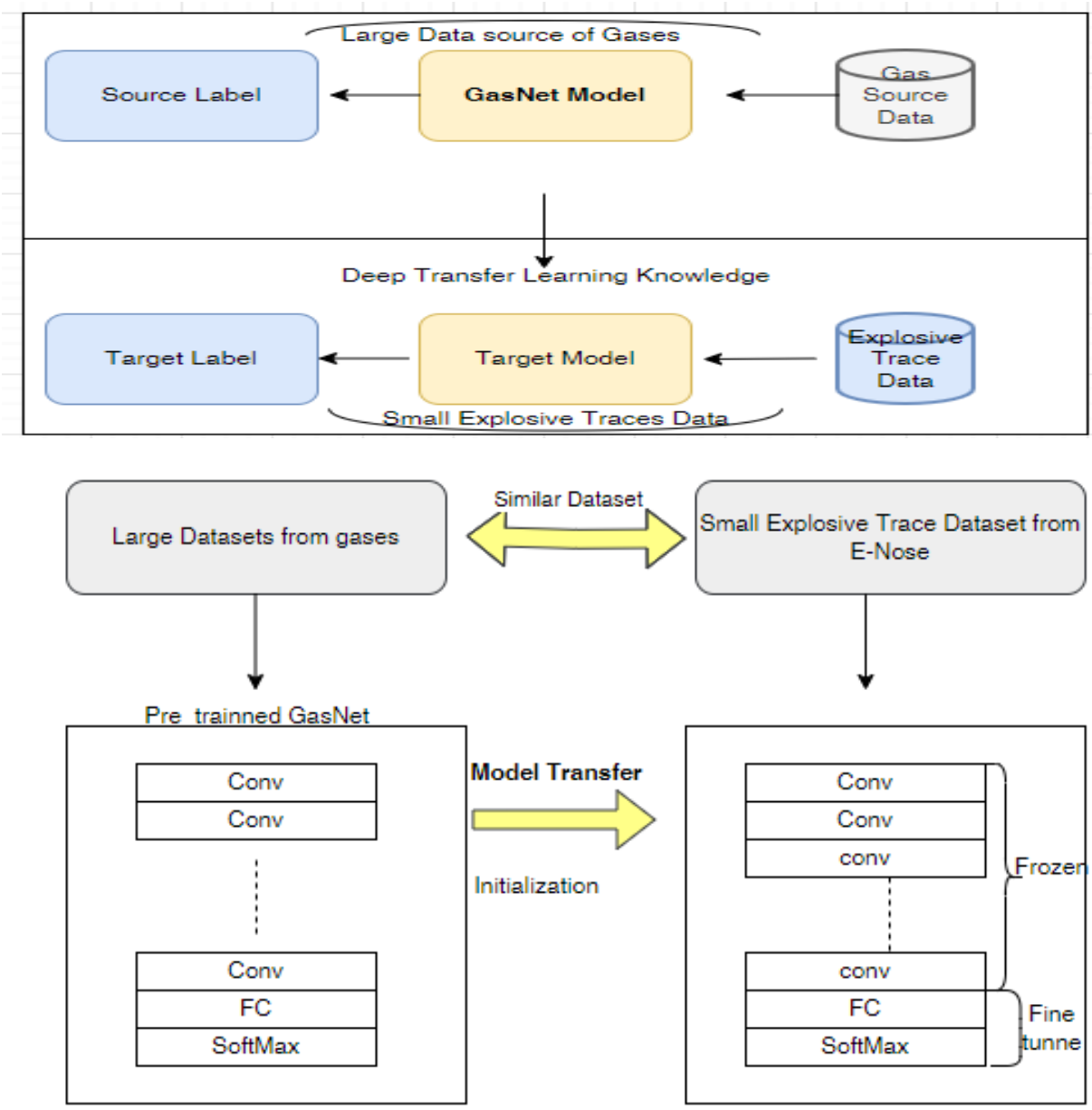


Figure 3.4: DTL Technique conceptual diagram

The source data are input into Deep Transfer Learning (DTCNN) for training based on GasNet DTCNN model. The model will be transfer to the target data to achieve the ETDTL model. Fine tuning will be done to achieve the best result in case of any variation between the source data and target data.

3.5 Data Collection

The dataset (Hossny *el tal.*, 2020) used in this research is the numerical data representing the concentration of gas traces. It is a vector of one-dimensional (non-spatial) data consisting of 1 X 5 features for a total of 69, 514 samples, with input features being C, N, O, H, and output feature being the target. The output state is either 1 or 0, where 1 represent a case when the combined concentrations of the input features suggest explosive trace, and 0 represent a case of non-explosive trace. However, since this dataset is on-spatial in nature, whereas deep learning and CNN in particular performs well on spatial or image data, it is appropriate to convert the source data to spatial or 2-dimensional data, a procedure which will map the vector samples into corresponding pixel equivalents as shown in Figure. 3.3, where the feature vector x is mapped or transposed to feature vector for each target sample.

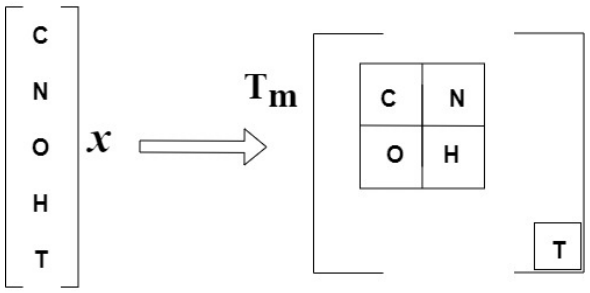


Figure 3.2: Data to Image Conversion

The conversion process follows a process that defined in the block diagram represented in Figure. 3.4, which shows that the process begins with obtaining the dataset, next is cleaning the dataset by scaling and normalizing the sample data. Feature engineering and visualization is performed to analyze the characteristics of the data. The fourth stage is to convert the vector data to spatial data before using it to tune the GasNet and eventually training a new model.

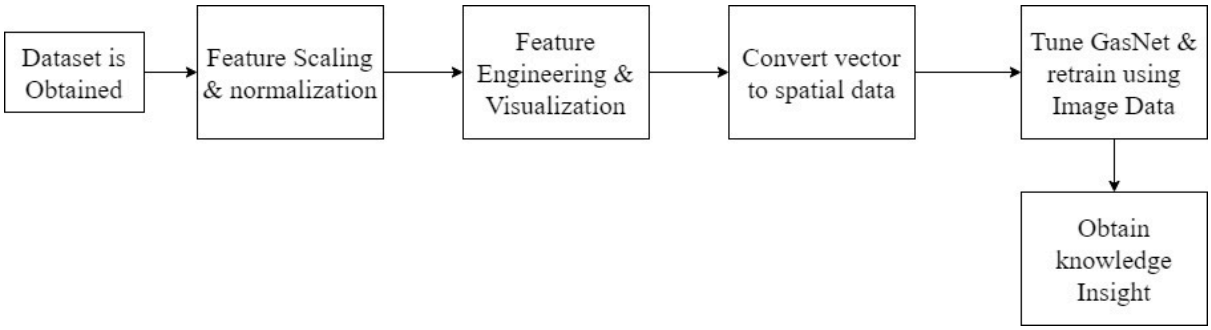


Figure 3.3: Block Diagram for Data Conversion

3.5.1 Data Normalization

The preprocessing approach starts with data normalization which involves the process of ensuring that all the feature data are within same range of 0 and 1. This process is important for ensuring that dataset does not overfit or underfit the model during training.

The method used to achieve this was Min – Max scaling technique, shown in equation 3.1 below.

$$X' = \frac{x - x_{min}}{x^{max} - x_{min}} \quad 3.1$$

Where x_{max} = the largest value of the feature

x_{min} = the smallest value of the feature

if x is minimum, $x - x_{min} = 0$ hence x' = 0

if x is maximum, $x - x_{min} = x_{max} - x_{min}$ hence x' = 1

if x is between max and min value, x' is between 0 and 1

3.5.2 Data Visualization and Balancing

The dataset used consists of 10, 000 data points or samples. The data were in two categories, namely explosive and non-explosive categories. Figure 3.4 shows the distribution of the dataset based on these categories.

The dataset was pre-processed after checking to determine if there were any missing value and dataset balanced between the two classes were done.

Figure 3.4 show the distribution of data categorize into explosive and non-explosive image that is as a result of numeric data conversion to create the image. During this process, the numerical values were read row by row, and using the row matrix to form a new 2x2 matrix. Each value in the matrix was used as a pixel value representing a shade of images.

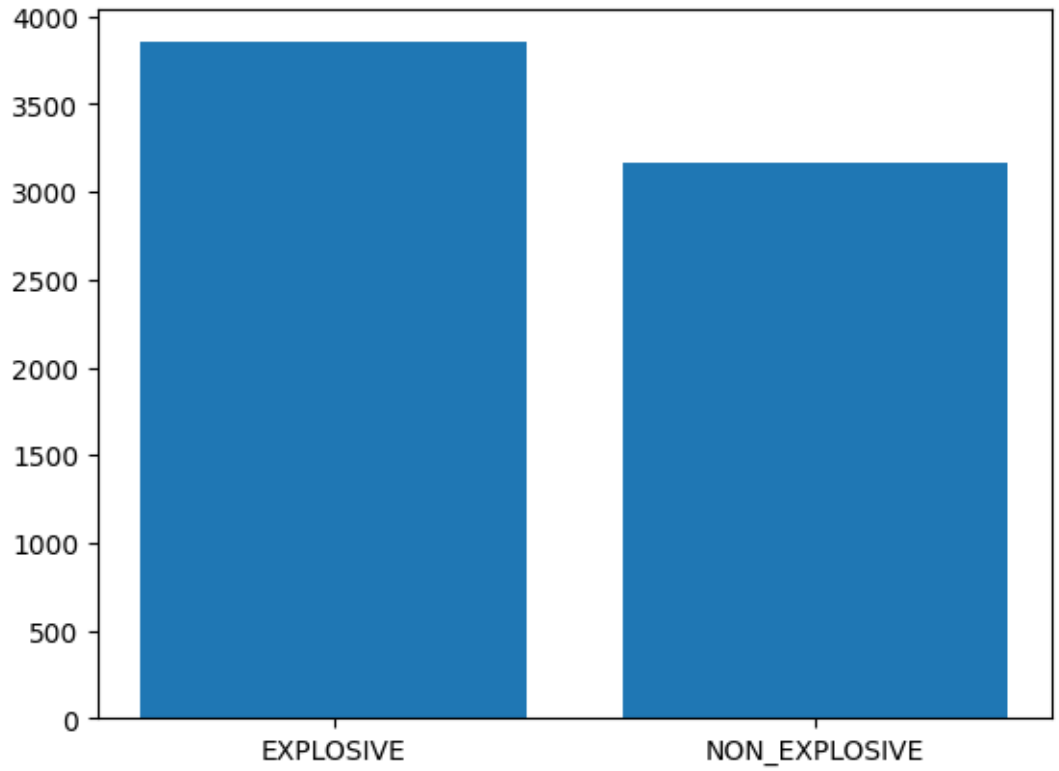


Figure 3.4: Distribution of the dataset categories

The distribution shows that the explosive category has a total of 5, 347 samples (53%), while the non-explosive category has 4,653 (47%) data samples. This distribution was a case of slight imbalance in the dataset, since there is the existence of majority and minority classes in the dataset. Such situation could lead to biased predictions and misleading accuracy. Therefore, this data must be balanced.

To create a balance in the dataset, the Synthetic Minority Oversampling Technique (SMOT) was used. This approach uses linear interpolation to create synthetic values of the minority class. Algorithm 1 below was used to implement the SMOT.

Algorithm 1:

Let the minority class set = A, such that $x \in A$

Loop*:*

Determine the k-nearest neighbor by computing the Euclidean distance between each x in set A

Set new x' between each nearest neighbor such $x' \in A$ *that where* $x' = x + rand\,(0,1) * |x - x_k|$. *Fix new point as x' along the lines segments of the neighbors*

Set N = N – 1

If N ≤ 0

Goto 5

Else Goto Loop

Stop

3.5.3 Data Conversion to 2D

During this process, the numerical values were read row by row, and using the row matrix to form a new 2x2 matrix. Each value in the matrix was used as a pixel value representing a shade of image as shown in figure 3.6 information. In this way, images were formed from numerical data. The output images were separated and stored as JPEG files into folders based on their respective classes. The two main folders created for this purpose were "Explosive" and "Non-Explosive". The code for the implementation is shown in appendix A in the appendices section. Moreover, the data were divided into training, testing and validation subsets. The training subset was 70% of the whole dataset, test subset was 20% and validation subset 10%.

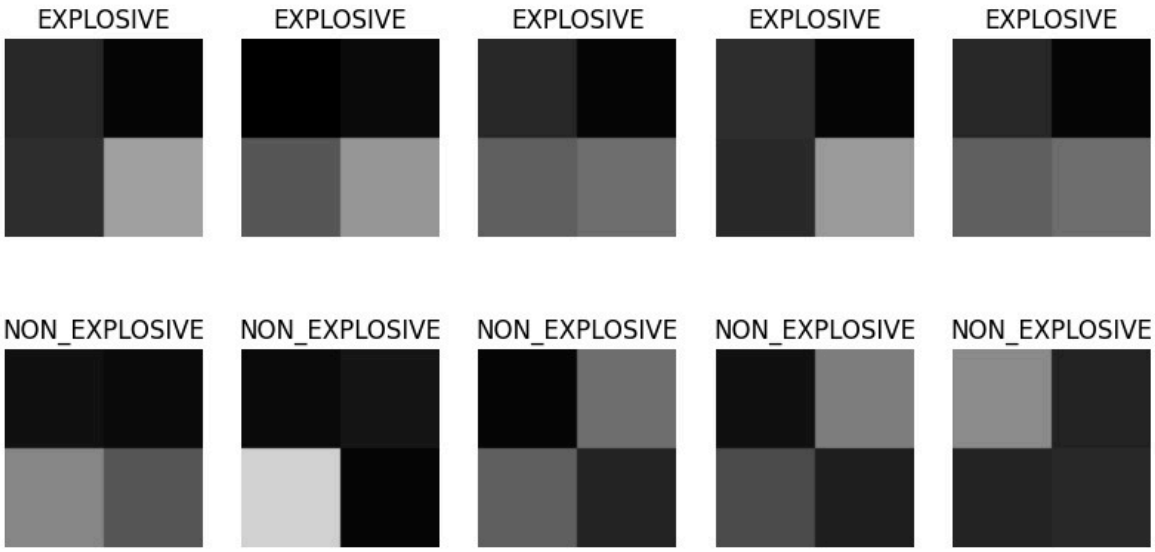


Figure 3.5: Explosive Trace Images and non-Explosive Images

This stage involves loading and preprocessing image data from the subsets. This was done by first defining variables for holding the image path where the images were stored. The code for achieving this is shown in Figure 3.7 below.

```
path='/content/drive/MyDrive/Dataset/'
print("Dataset path:  ")
print(os.listdir(path))

train_dir=path+'Train'
test_dir=path+'Test'
val_dir=path+'Val'
print("Sample path:  ")
print(val_dir)
```

Figure 3.6: Code for Data to Image Conversion

When the images were loaded into the variables, the dimensions were further reshaped to ensure that they all maintain the same size. This was achieved using the following code in python.

```
input_shape = (24, 24, 1)  # Adjust dimensions based on your dataset
```

This means that all the images will maintain height and width pixel dimensions of 24 x 24, 1 channel since the images are already in gray scale.

3.5.4. Image Data Augmentation

Data augmentation is process that ensured that our model generalized well. The **ImageDataGenerator** class in keras was used for this purpose. The process includes a series of random transformation of images such as rotation, flipping, zooming, cropping and brightness/contrast adjustment. This stage was achieved using the code snippet shown in figure 3.8 below, while the full code for the augmentation was shown in appendix E

```
# Data augmentation for the training set (optional but recommended)
train_datagen = ImageDataGenerator(
    rescale=1./255,      # Normalize pixel values to [0, 1]
    rotation_range=40,  # Randomly rotate images
    width_shift_range=0.2,  # Randomly shift images horizontally
    height_shift_range=0.2,  # Randomly shift images vertically
    shear_range=0.2,  # Shear transformations
    zoom_range=0.2,  # Randomly zoom images
    horizontal_flip=True,  # Randomly flip images horizontally
    fill_mode='nearest'  # Fill mode for pixel values outside the image
)
```

Figure 3.7: Image Data Augmentation code

The code show that our images were randomly rotated by 40 degrees, scaled by a factor of 1/255, horizontally and vertically skewed to 0.2, and zoomed by a factor of 0.2.

3.5.5. Convolution Neural Network Development

In this stage, the CNN model was developed with the development phase following the general structure of the CNN architecture. The stages involved the following layer development:

1. Convolution layer design
2. Fully connected layer design
3. Design of the Output Layer

3.5.5.1. Design of the Convolution Layer

The convolution layer consists of the following:

1. Image feature map or image matrix, X, which is a 2x2, which was scaled by padding to 3x3 matrix (2D)
2. A filter, f, a 2x2 matrix.

The operation performed at this layer therefore is the convolution, Z(2x2) of X and f, which is the sum of the element-wise product of X and f, and this can be expressed as:

$$Z(2,2) = \Sigma X(3,3) * f(2,2) \quad 3.2$$

Our convolution layer with 3 D-sub layers was developed using keras in the code snippet presented in Figure 3.9

```
layers.Conv2D(32, (2, 2), activation='relu', input_shape=input_shape),
layers.MaxPooling2D((2, 2)),
layers.Conv2D(64, (2, 2), activation='relu'),
layers.MaxPooling2D((2, 2)),
layers.Conv2D(128, (2, 2), activation='relu'),
layers.MaxPooling2D((2, 2)),
```

Figure 3.8: Code for Convolution Layer Development

The first 2D convolution layer was a designed with 32 filters, each being a 2x2 matrix filter, which uses the Rectified Linear Unit (ReLU) activation function. The output of the first convolution (Conv) layer was passed through 2x2 Max Pooling operation, before being fed to the next Conv layer. The second Conv layer had 64 2x2 filters, with ReLU activation function. The third Conv layer had 128 2x2 filters also with ReLU activation function.

3.5.5.2. Design of the Fully Connected Layer

This is a neural network layer and can only work with 1D data. This implies that the output of the last Conv layer, which is a 2D must be converted into a 1D image by flattening as shown in Figure 3.10, and was now to be fed into the fully connected layer. In the fully connected (FC) layer, linear and non-linear transformation operations were performed on the 1D data fed into it.

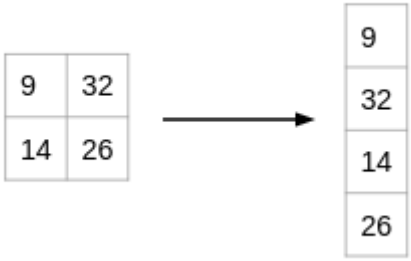


Figure 3.9: Conversion of 2D to 1D

The linear transformation operation is represented by equation 3.3 below.

$$Z = w^T \cdot X + b \qquad 2.3$$

Were

$\boldsymbol{X}$ is a vector of the image feature extracted from Conv layers

$\boldsymbol{w}$ is a 4x2 the matrix of weight (a matrix of randomly assigned values)

$\boldsymbol{b}$ is a vector of biases (a constant value)

The FC had 2 neuron to linearly transform 4 data points in the X image vector. Therefore, Z was given as

$$Z = \begin{bmatrix} w_{11} & w_{21} & w_{31} & w_{41} \\ w_{12} & w_{22} & w_{32} & w_{42} \end{bmatrix} \begin{bmatrix} D_1 \\ D_2 \\ D_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \qquad 3.4$$

During the non-linear transformation, an activation function was chosen for the output of the FC. Sigmoid function in equation 3.5 was the best choice at this stage because we are dealing with binary classification.

$$f(x) = \frac{1}{1 + e^{-x}} \qquad 3.$$

The final task during the model development was to determine the method of optimization, a process that was used to update the learning rate of the model to ensure that all computations converge correctly. Adam gradient decent defined in equation 3.6 was used.

$$\theta_2 = \theta_1 - (\alpha \times gradient\ parameter) \quad 2.6$$

Where

θ_1 is the New parameter

θ_2 is the old parameter

α is the learning rate (a constant that determines the amount of change to be made to old parameter)

Gradient is the change in classification error with respect to parameter

The code snippet that was used for the implementation is illustrated in Figure 3.11

```
layers.Flatten(),
layers.Dense(128, activation='relu'),
layers.Dense(2, activation='sigmoid')  # num_classes is the number of output classes
```

Figure 3.10: Model Optimization Code

3.6. Approach and Technique(s) for Explosive Trace Detection Using Deep Transfer Learning

The approach and technique used in this research is experimental approach, in which experimental model of machine learning algorithm based on Deep Transfer Learning was developed for the purpose of detecting explosive traces by means of classification. Transfer Learning (TL) is a machine learning method that transfers the skill used in learning a task to another scenario of learning a different task. Deep Transfer Learning (DTL) is a TL that is based on the deep neural network architecture.

In this research, explosive traces were detected using Artificial Intelligence (AI) model. Deep Transfer Learning model was developed from a base model known as GasNet from (Barrera et al., 2020) which was developed from Deep Convolution Neural Network with 38 layers. Out of these number of layers, 6 inner (deep) layers were frozen, reconstructed (tuned) and trained using Explosive Gas concentration dataset from UCI website. To classify explosive gas within

an area, sensor are deployed to detect the concentration of Carbon (C), Hydrogen (H), Oxygen (O), and Nitrogen (N) gasses, which are combined and sent to the newly trained model. The model then uses the learned knowledge to classify the gas combinations as either explosive or not.

3.6.1 Transfer Learning Model Formulation for Explosive Trace Detection

Transfer learning is based on the theory of Identical Element (IE), which state that transfer of learning will take place if the two tasks to learn from have identical features (Campbell et al., 2006). This suggests that in the case of this research work, that GasNet and the proposed model of this research should have some sort of identical features in their input dataset. This implies that the output of output function may be different from each other.

The notation representing TL in its general context is that if a domain, D is given by

$$D = \{X, P(X)\} \quad 3.7$$

Where X = Feature Space

P(X) = Marginal Probability Distribution

X = $\{x_1, x_2, x_3 \ldots, x_n\} \in X$

In the context of this research, x_1, x_2, x_3, x_4 will represent C, H, N, and O, and hence will be represented as x_C, x_H, x_N, x_O

This therefore implies that, given an explosive classification label, Y, can be determined by the predictive function, $f(.)$, which can be learned from the training dataset $(x_i, y_i) | i \in X$, where $x_i \in X$, and $y_i \in Y$. This implies that $f(x_i) \equiv P(y_i | x_i)$ is the conditional probability of detecting an explosive gas from a given set of x_C, x_H, x_N, x_O

The conceptual framework is presented in figure. 3.2. The framework shows the various layers involved in the development of the model, which achieves the first objective of this research. The base model is a DCNN model that have been previously trained with very large dataset, for the detection of gases in an e-nose application. The model was tuned by reconstructing the inner or deep layers by freezing 6 layers and replacing them with new ones for the application of the task at hand. This process is also known as feature extraction. The final stage is the training of the new model by adjusting and testing on new data.

In this framework, the base model is first tuned by freezing 6 inner layers and reconstructing them by introducing new layers since $P(Y_t | X_t)_{Explosive} \neq P(Y_s | X_s)_{GasNet}$).

3.6.2 Transfer Learning Model Development

The developed CNN model in section 3.5.5 was modified at the output layers so that data generated by a simulation model were fed into it for further training and to demonstrate a virtual deployment environment (see section 3.x1) for areal-based explosive testing. This setup was necessary to be able to test the model with real-time data for the sake of validating the results of the model's performance metrics.

The procedure in developing the DTLETD model is represented in the activity diagram shown in figure 3.12 that shows the data generation stage, development of new model and new model validation. This process the following stages:

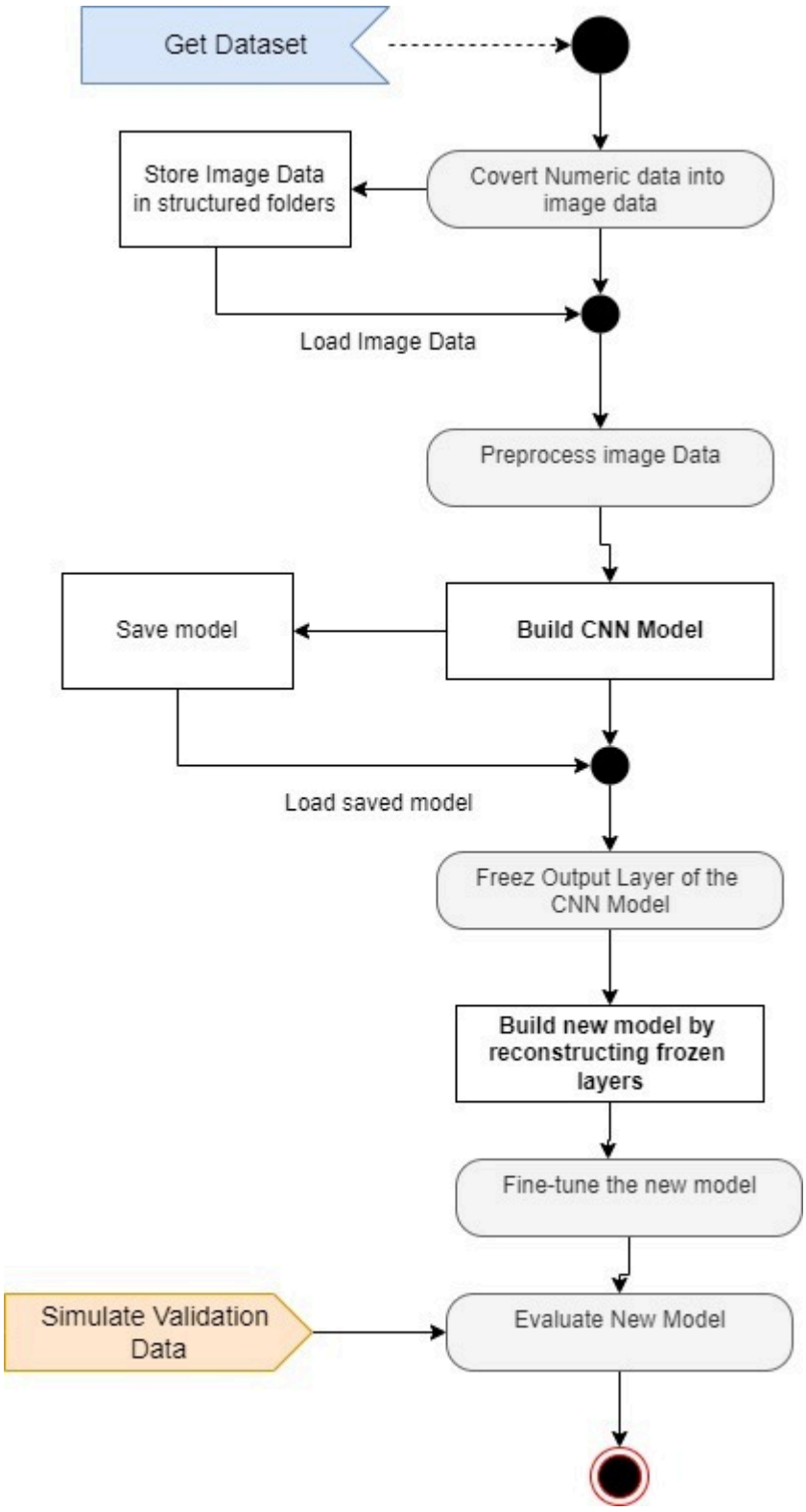


Figure 3.12: Transfer Learning Activity Diagram for Explosive Trace Detection

As it can be seen in figure 3.12, a set of simulated data were generated using the simulation model in section 3.7. The generated data were in the range and format of the dataset used in

training the base model. That is data were generated each for C, N, O and H features respectively. A set of 20 samples of simulated data were generated as shown in table 4.1.

After storing the simulated data in a table, the pre-trained CNN model was loaded the Convolution Layer of the CNN model was frozen to prevent that layer from being affected by the modification during the development of the transfer learning model. The code section responsible for this is given in figure 3.13

```
for layer in base_model.layers[:-5]:
    layer.trainable = False
```

Figure 3.13: Code for Freezing the Convolution Layer

This stage was for the development of the remaining layers of the CNN model in order to develop a new model, while retaining the components of the frozen layer. This was achieved using the code snippet in figure 3.14. The code has the first line for initializing a model variable, followed by the line that called the base-model, which is the pre-built CNN model. After that was then added, a new output layer (flatten layer). A new dense layer with 128 units was added, and each fully connected to the flattened layer through ReLU activation function. This was followed by a dropout layer with a dropout rate of 0.5. Finally, an output layer was added with a single unit of sigmoid activation function.

```
model = models.Sequential()
model.add(base_model)
model.add(layers.Flatten())
model.add(layers.Dense(128, activation='relu'))
model.add(layers.Dropout(0.5))
model.add(layers.Dense(1, activation='sigmoid'))
```

Figure 3.4: Code for creating new model

During this stage, the newly developed model was trained using the validation data samples, which was about 10% of the base dataset. The code snippet is presented in figure 3.15 and the full code is shown in appendix C. The code typically represents steps for the model to learn from new data samples while retaining its knowledge of the previous experience. The code also specified the hypermeters, such as learning rate, batch size, and epochs.

```
history = model.fit(
    train_generator,
    steps_per_epoch=train_generator.samples // batch_size,
    epochs=epochs,
    validation_data=val_generator,
    validation_steps=val_generator.samples // batch_size,
    callbacks=[lr_scheduler]
)
```

Figure 3.11: Code for fine-tuning the new model

The next stage was the evaluation of the newly built model. The fine-tuning accuracy score and the AUC were determined and recorded. The system was further tested using the simulation gas data, and the test accuracy score and AUC were recorded as well. Graphs of the system training validation losses against the epochs as well as losses against learning rate were also plotted and presented in figure 4.8 and 4.9 respectively.

3.7 System Deployment and Testing

The block diagram of figure 3.16 represents the implementation model of the area-based explosive trace detection system. This set up is meant to generate data for the developed model for validation as shown in the design framework in figure 3.1. The diagram has four major sections, which include the Input Unit, Edge Interface Unit (EIU), Cloud Intelligent Unit (CIU), and the Output Unit.

3.7.1 Description of the Input Unit

The input unit consists of an array of three sensors from the MQ series. The sensors are responsible for the sensing of the explosive traces by detecting certain characteristics of the various trace components in the deployed environment. Our input trace elements are the C, N, O, and H gases as shown in figure 3.1. Therefore, the categories of MQ series sensors used were such that could detect the presence of these trace elements. For instance, MQ-7 was used for detecting CO from which C was detected, MQ- was used for the detection of Hydrogen gas, while MQ-135 was used for the detection of the Oxygen and Nitrogen gases.

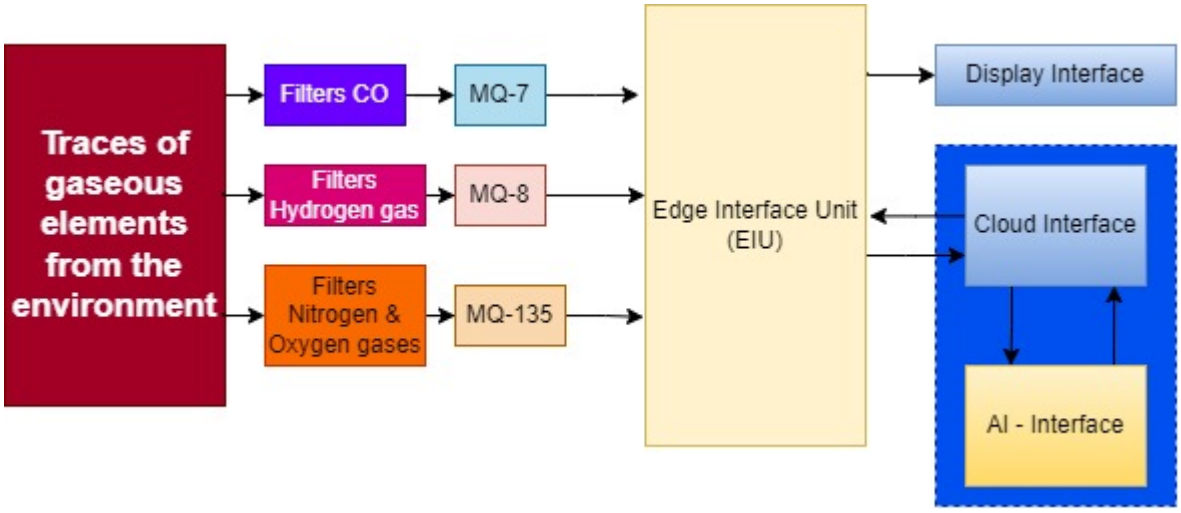


Figure 3.12: Block Diagram of the Implementation model

The characteristics features of these sensors are presented in table 3.1. The sensors simply converts any gas trace within its range of detection into an equivalent analogue signal, which is measurable as a voltage range at the output pin of the sensors. The output analogue voltage was fed into the EIU for further processing tasks.

Table 3.1: Characteristics of the Sensor used

Characteristics	**Sensors**		
	MQ-7	**MQ-8**	**MQ-135**
Operating voltage	+5V	+5V	+5V
Sensitive to	CO	Hydrogen	NOH, NH3
Analogue output range	0 – 5V	0 – 5V	0 – 5V

3.7.2 Description of the Edge Interface Unit (EIU)

The EIU represents the processing unit, which receives the input signals from the sensor array. Firstly, the EIU performs the analogue to digital conversion on all the input signals. Secondly, the EIU separate the input signals into various components before compressing the signals using 50th term averaging methods to reduce signal noise due to false trigger. Thirdly, the compressed digital signal was packaged transmitted to the cloud using HTTP protocol and Thingspeak cloud. The result of the out was collected. It should be noted that the out graph generated on thinkspeak has a 10^{-1} multiplier, this is to accommodate thingspeak value range.

3.7.3 Cloud Intelligent Unit (CIU)

At the CIU, each signal component was collated for storage, visualization and further classification processing using the already built Transfer Learning Model.

Communication between this unit and the EIU is full duplex mode to allow for feedback to the EIU and for effective communication of the result of classification to the output unit. The output unit is an LCD module, which was used primarily for the display of the classification result at any selected instance.

3.7.4 System Circuit Development and Testing

For a real deployment and testing for the detection of explosive traces, circuit in figure 3.17 was developed following the block diagram in figure 3.16 as described in section 3.7.1 above, the input interface was realized using the MQ series gas sensors. The EIC was realized using Arduino Uno with in-built ESP wifi module. The wifi module enabled seamless transmission of data to the cloud. The output of the MQ-8 was connected to the analogue input of the Arduino board (A0), MQ-7 was like-wise connected to the A1 pin and MQ-135 was connected to the A2 of the Arduino board. During the simulation testing, the potentiometers, RV1, RV2, and RV3 were varied to allow the MQ sensors to generate random output values. Each set of generated output values were processed accordingly and transmitted to the cloud server using the HTTP protocol. The Arduino code for proteus simulation is shown in appendix F.

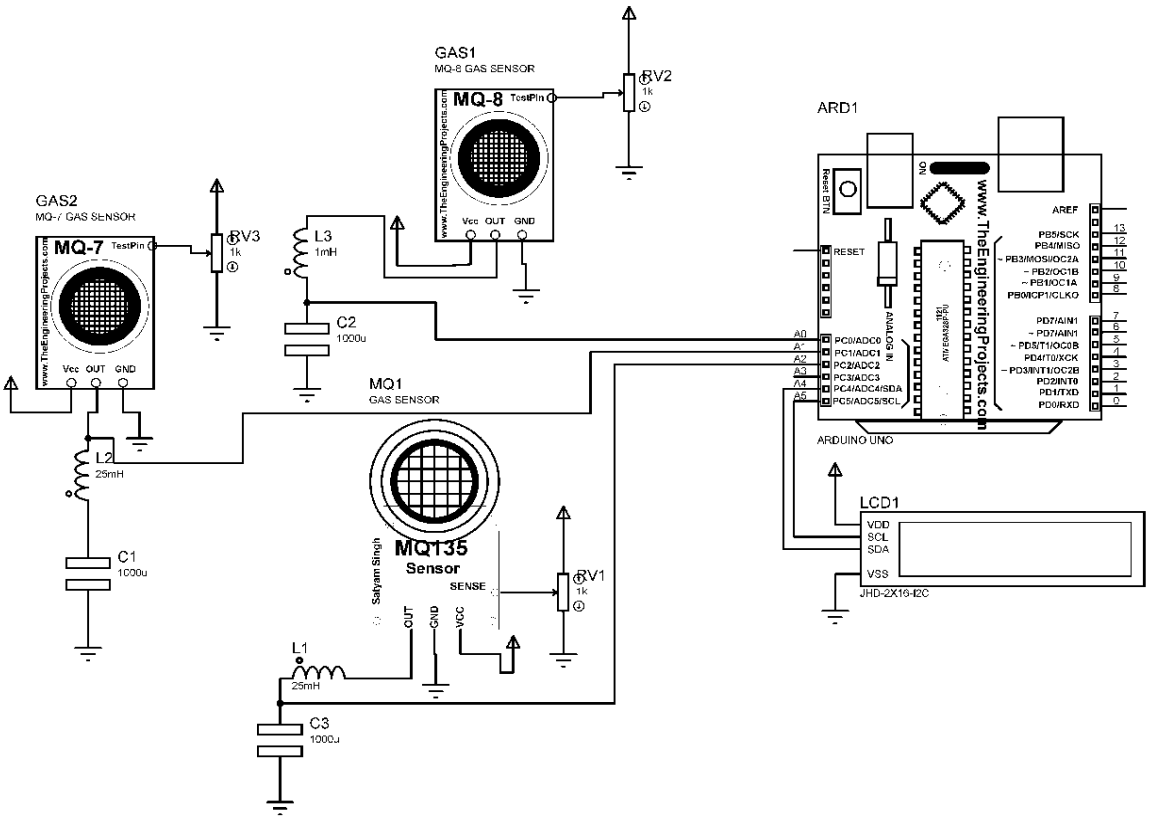


Figure 3.13: Circuit diagram of the Implemented system

3.8 Description of Performance evaluation parameters/metrics

The acquired dataset is split into a training dataset and a test dataset, with 70% of the data allocated for training and 20% for testing, while 10% for validation. This hold-out method is commonly used to train DNN(Nguyen et al., 2021). The training dataset used to train the model utilizing a three-fold cross-validation approach. This approach helps to prevent the model from becoming biased by analyzing the evaluation metrics on different folds of the data. The test dataset will be employed to assess the accuracy of the trained model on sensor dataset for the trained model to make predictions and comparing those predictions to the actual values.

The trained model accuracy is obtained using Equation 3.7 and Equation 3.8, with the test dataset. Where TP is the True Positive, FP is the False Positive, FN is the False Negative, and TN is the True Negative. TP refers to an accurate prediction of a positive explosive trace, while TN refers to an accurate prediction of a negative explosive trace. FP occurs when a negative explosive trace is predicted as a positive one, and FN occurs

when a positive explosive trace is predicted as a negative one. The F1-score was computed using Precision (Equation 3.9) and Recall (Equation 3.10).

Confusion matrices are an effective tool for evaluating the accuracy of a classification model as they provide a more detailed breakdown of the model's performance than a simple accuracy score. Specifically, a confusion matrix tallies the number of true positives, true negatives, false positives, and false negatives for each class or category. By examining these values, we can calculate a range of performance metrics such as precision, recall, and F1 score, which provide a more balance and informative picture of the model's accuracy. Additionally, by evaluating a model's accuracy using a confusion matrix, we can identify which categories or classes are being misclassified most frequently and adjust our model accordingly. This can help us optimize our model's accuracy for specific applications and ensure that it is performing effectively in real-world scenarios.

We will also use Area under Curve (AUC) to check whether the performance score is a true representation of the accuracy. The higher the AUC the better the system performance and that is a better way to check the system robustness.

The validation of the model, the model was tested on Explosive trace dataset from other sensor network but in this virtual simulated network generated a live explosive data to determine its performance. The performance is compared with other traditional machine learning methods and other DL model in terms of accuracy and AUC. This proposed method is achieved using Python software, Keras with Tensorflow as the backend. The trained GasNet model will then be imported from keras to be ran on core i5, 8GHz Laptop, using NVIDIA GTX960 GPU.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100 \quad 3.7$$

$$F1 - score = \frac{2 \times precision \times Recall}{Precision + Recall} \quad 3.8$$

$$Precision = \frac{TP}{TP + FP} \quad 3.9$$

$$Recall = \frac{TP}{TP + FN} \quad 3.10$$

Apart from these metrics the Receiver Operating Characteristic Area under Curve (ROC-AUC) was also used to test the level of accuracy of the model

AUC = Area under the ROC curve

The ROC curve is a plot of the True Positive Rate (TPR) on the vertical axis, given as

$$TPR = \frac{TP}{TP + FN} \quad 3.11$$

Against the False Positive Rate (FPR) on the horizontal axis, given as

$$FPR = \frac{FP}{TN + FP} \quad 3.12$$

3.9 Tools used in Implementation

The tools that was utilized to realize this research include the following:

a. Core i5, 8ghz personal Computer (PC) that was used to carry out all the software operations
b. Draw.io software was used to design the framework, flowchart and diagrams.
c. Google Collab was used to deploy the machine learning models, it was able to handle the computational load of classification.
d. Python software was used to write the codes
e. Keras with Tensorflow as the backend was used to train the model before it was exported to be ran on the PC with.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Preamble

The results for the various experiments are presented in this chapter, firstly with the system evaluation, the result for the proposed base model of CNN and the DTLETD. The result was discussed after its presentation and was finally compared with other known models all in accordance with the research questions to be addressed.

4.2 System Evaluation

In this study, the results are presented both Deep Learning model and Deep Transfer Learning Model. The Deep Learning model validation accuracy serves as a measure of how well the model perform on the explosive trace data. The performance of the model will be evaluate when transfer has not occur. The second evaluation will be on the Deep Transfer Learning model, where the model training loss will be evaluated together with the performance rate on limited data. The rate at which the developed model learn will also be evaluated. By presenting the results in this way, we can demonstrate the effectiveness and generalization capability of DTL model and provide insight into how it can be further improved in the future. Overall, emphasis is placed on learning rate and thorough evaluation to ensure that the model is performing at a high accuracy and is ready for deployment in real-world scenarios on edge devices.

Deep learning models have become increasingly popular and powerful in recent years, providing accurate predictions, and helping to enable many innovative applications.

4.3 Results presentation and Analysis for Explosive trace Detection using CNN

The results obtained from data preprocessing to the point of model deployment is presented in this session with the corresponding analysis of the results.

4.3.1 Explosive Data Preparation Results

This results in this section shows explosive and non-explosive data set and the data conversion. Figure 4.1 is the explosive trace and non-explosive trace data distribution used in this work while figure 4.2 shows the data conversion result of the same.

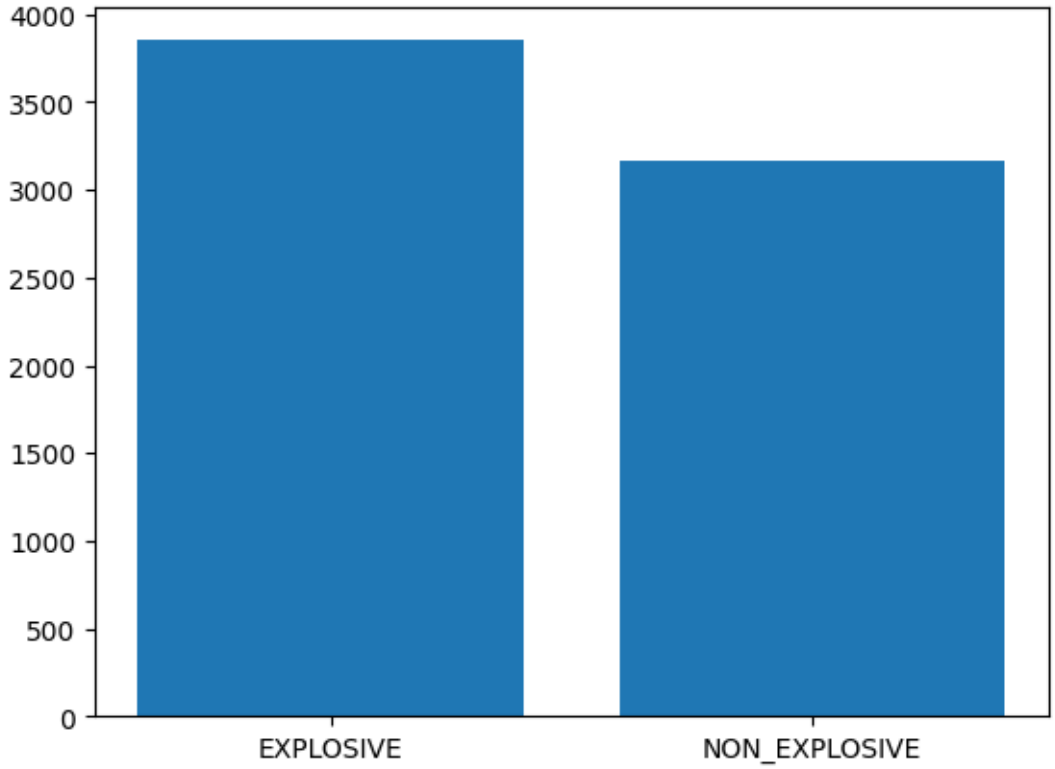


Figure 4.1: Explosive and Non-Explosive Dataset Distribution result

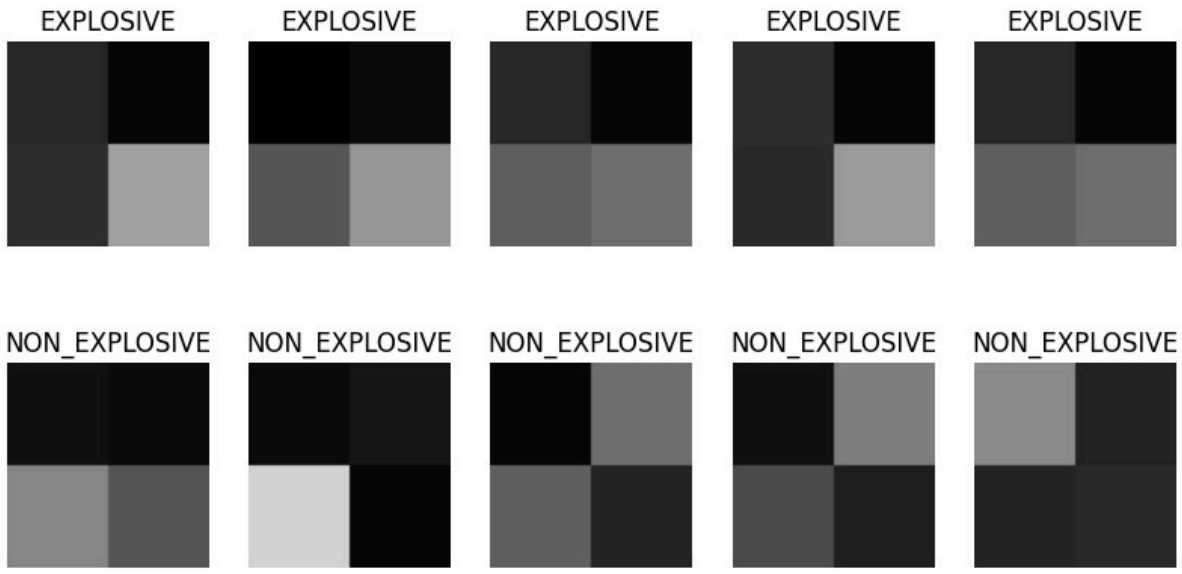


Figure 4.2: Result of Explosive and Non-Explosive 2D Data

4.3.2 The result and Analysis of CNN on Explosive Trace Detection

The result of the model deployed using python 3.10 is shown in Figure 4.3, it shows the graph of loss against the epochs. The result show that during each epoch, the losses in the developed model was inversely proportional to the epochs. Meaning that error that would produce

misleading production was reduced sufficiently. This confirms that the model performed well with the dataset. The validation was done using validation dataset.

The result of the training performance evaluation is shown in Figure 4.4, it presents how the model performed during training. Accuracy was used as the metric of evaluation. From that graph, we see that during each iteration, the accuracy of the model was increasing and achieved 98.2% accuracy score.

The confusion matrix was used to evaluate the system performance during testing. This plot is shown in Figure 4.5. The result shows that all the 32 samples used for the test were correctly classified. Out of that number, 18 samples were correctly classified as explosive, while the remaining 14 samples were also correctly classified as non-explosives. Also the ROC curve in Figure 4.6 also confirms that the model performed very well. For both classes, the model archived an area under curve (AUC) value as 1. This is the highest any model can achieve. These result and those of other metrics are presented in table 4.1 and appendix G shows the calculation.

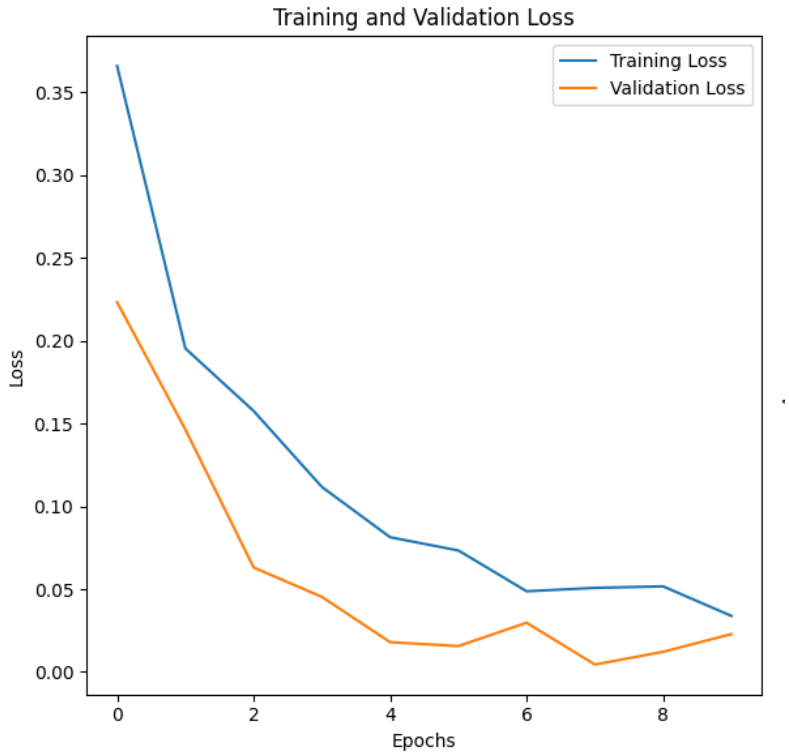


Figure 4.3: Training and Validation Losses Result

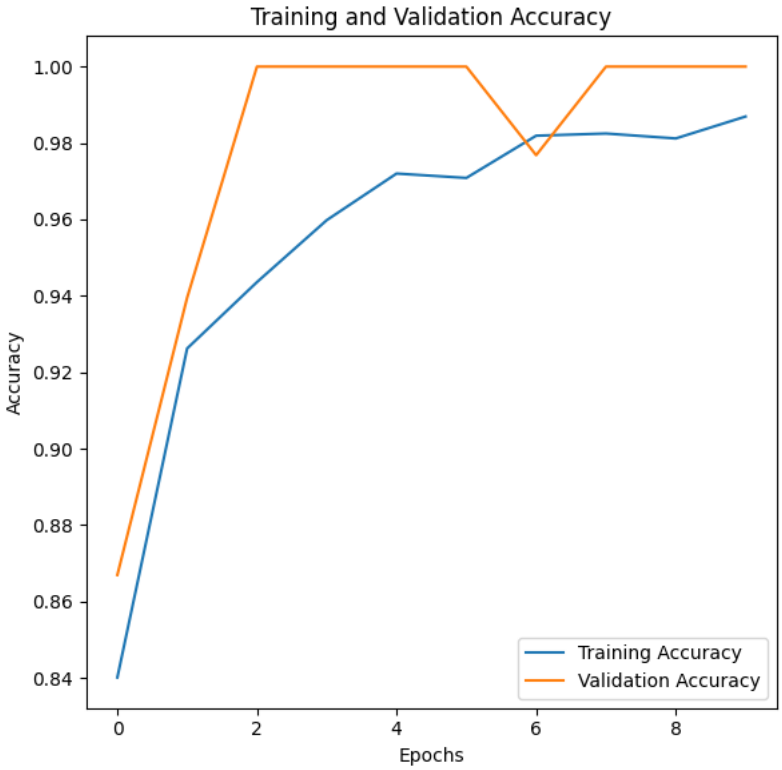


Figure 4.4: Graph of Accuracy against Epochs Result

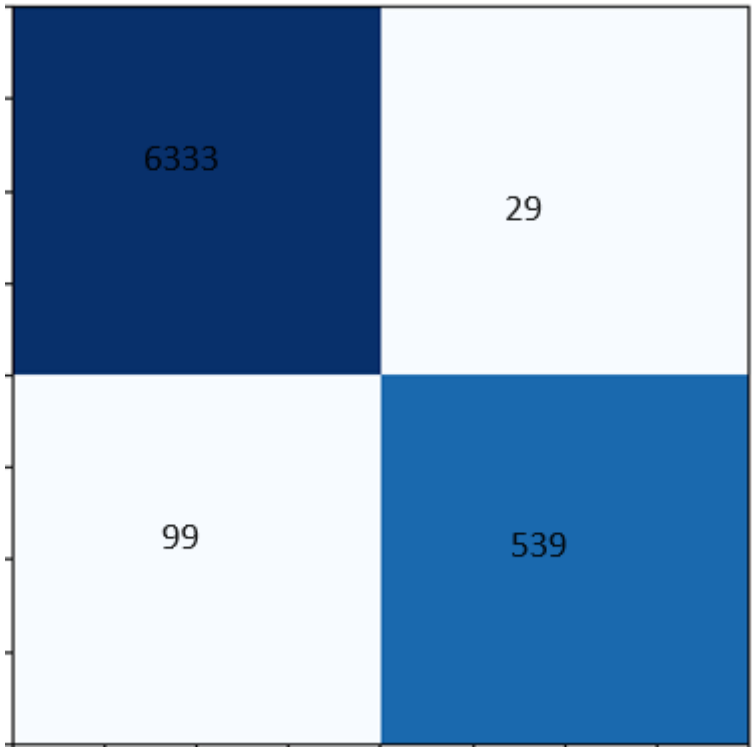


Figure 4.5: Confusion Metrix of the CNN after Training with 7000 data points

Table 4.1 Other performance Metrics of the CNN Model

ACC	PRECISION	RECALL	F-1 SCORE	SPECIFICITY	FPR	TPR	AUC
0.982	0.985	0.985	0.985	0.949	0.051	0.985	1.00

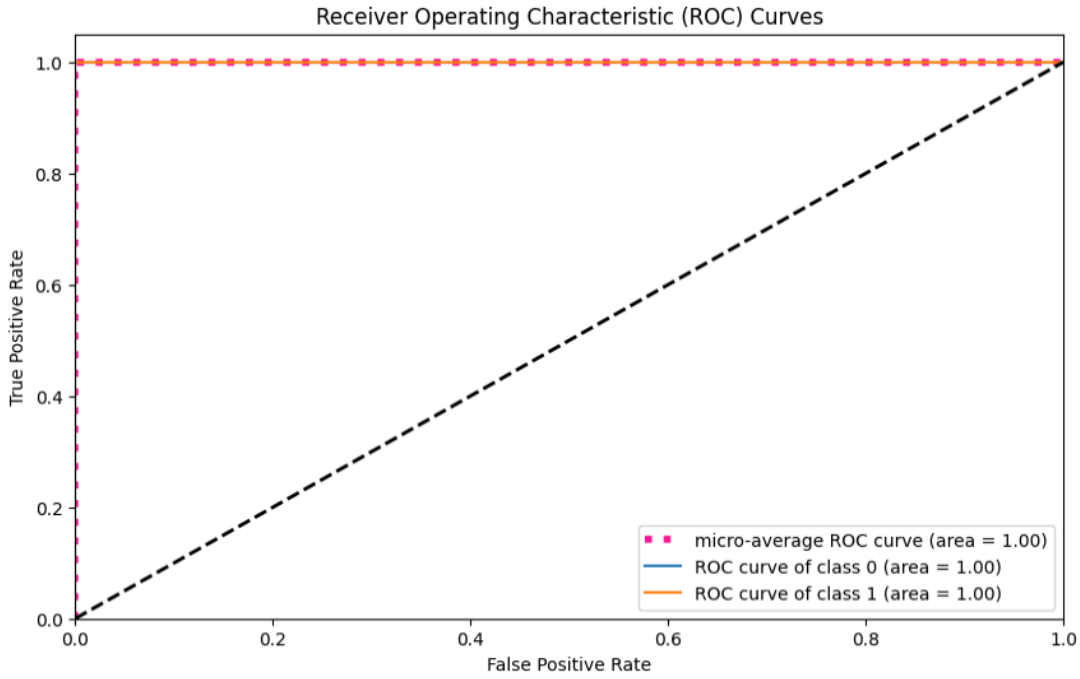


Figure 4.6: Receiver Operation Characteristic Curve (ROC)

4.3.3 Result of Deep Transfer Learning Model for Explosive trace detection

The simulation model described in section 3.6.1 was first used to generate some random new samples of data resembling those of the original dataset. The generated samples were stored in table 4.1. These set of data was used for the validation testing of the model, by being used as input to the transfer learning model. The prediction yielded the target values in each case.

The developed transfer learning model was trained with only 3 epochs and the graph in Figure 4.8 was generated. The graph is a plot of training losses and validation losses against the epochs. The result show that the training losses dropped sharply from 0.15 to 0.08 during the iteration of the first epoch and converged before the second epoch. The validation losses also dropped from less than 0.05 to 0 within the first epoch and it remained 0 during the second epoch.

The result reveals the following significant achievements:

1. The transfer learning model took less time (about 92 seconds) to train against a training time of about 1287 seconds used to train the CNN model.
2. The transfer learning model converged faster than the with nearly zero losses for both training and validation

The Convergence of learning rate of the Transfer Learning Model is shown in appendix D while Figure 4.9 is a graph of the training and validation accuracy against the epochs is presented. During the iteration of the first epoch, it can be seen that the training accuracy already reach 99.7%, while the validation accuracy remained at 100% from the iteration of the first epoch. Confusion matric presented in Figure 4.7 is also the performance report obtained during this time. Other results were the reports of other performance matrices shown in table 4.2.

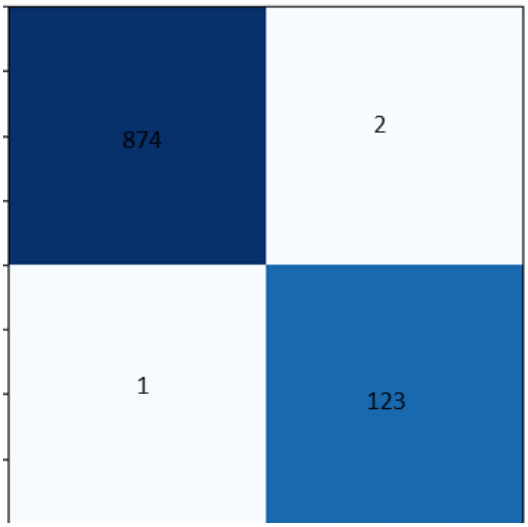


Figure 4.7: Confusion matrix of the DTLETD after Training with 1000 data points

Table 4.2: DTLETD Performance Metrics

ACC	PRECISION	RECALL	F-1 SCORE	SPECIFICITY	False Positive Rate	True Positive Rate	AUC
0.997	0.999	0.999	0.999	0.984	0.016	0.999	0.89

This result confirms that the transfer learning model adjusted very quickly with the dataset to achieve very high and stable performance with the few data samples and small epoch size.

The graph in Figure 4.10 represents the effect of the tuning of the learning rate (lr) hyperparameter on the training losses. The lr was kept as small as 10^{-3} at the start of the training. It was gradually increased as the training progresses. The effect of increasing this hyperparameters was that the losses got smaller until it converged at lr = 1.25×10^{-3}.

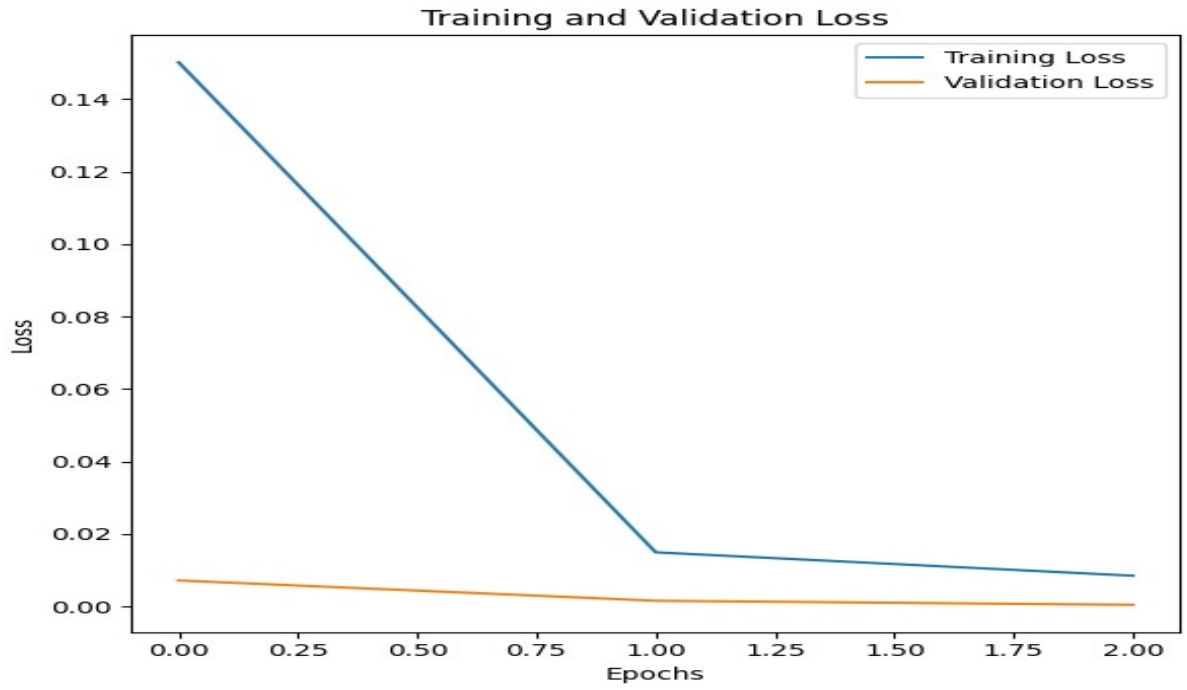


Figure 4.8: Transfer Learning Training and Validation Losses against Epochs

Figure 4.9: Transfer Learning training accuracy and validation against Epochs

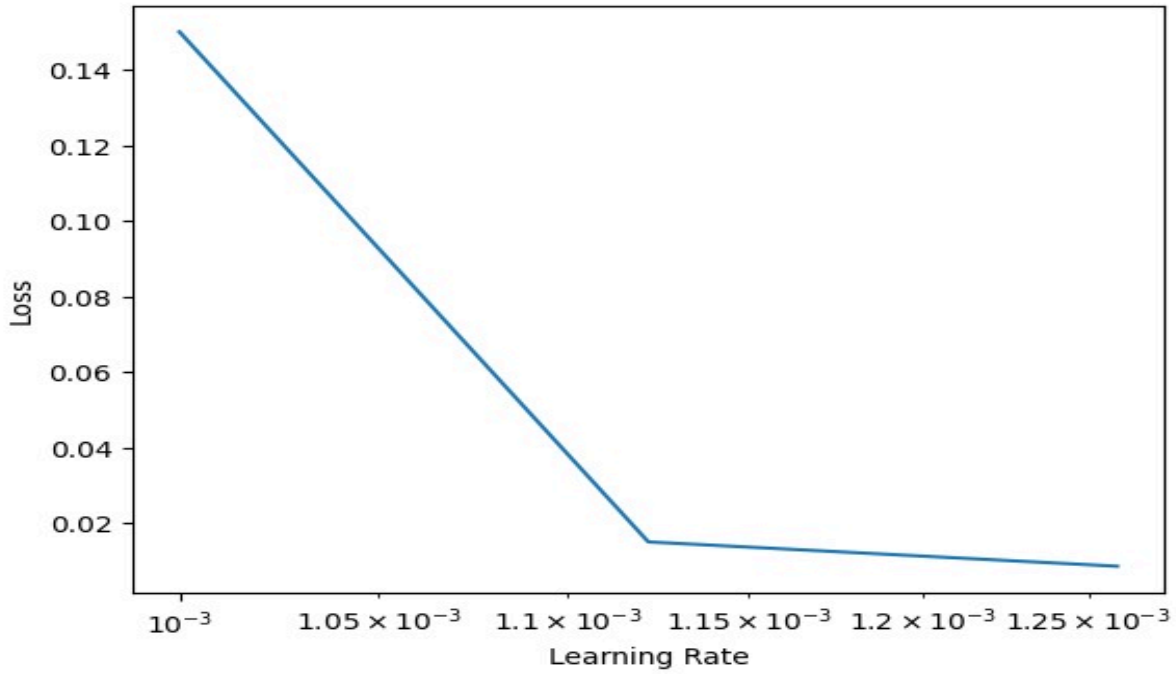


Figure 4.10: Effect of Tunning on the learning rate

4.3.4 Results of the Simulation Model

The simulation model of the system deployment was completed using Proteus and Thingspeak cloud. The circuit was setup as shown in Figure 3.15. During simulation, the system communicated with the Thingspeak cloud via a Wi-Fi connection interface and show the data transmission progress illustrated in Figure 4.11.

Other results of data transmission to the cloud are shown in Figure 4.12 (a) through 4.12(d). This shows the various values of the explosive traced detected by simulation model over a period of about 30 minutes with 10^{-1} multiplier. The range of values were between 0 and 1 as used in the original dataset. These simulated data were used also to further validate the transfer learning model, which yielded the results shown in table 4.1, showing that the model achieved an average prediction accuracy of 99.7%, with an average AUC value of about 0.89. The system also yielded a precession of 96%. This result shows that the developed transfer learning model could still produce high performance metrics values after deploying different data on it.

Table 4.3: Simulated Sample Data of Explosive Trace

C	N	O	H	Target
0.677983	0.164722	0.516603	0.589594	1
0.528973	0.694318	0.860471	0.136377	1
0.053052	0.356432	0.106087	0.519277	1
0.748067	0.553447	0.365401	0.520955	0
0.032338	0.981841	0.780361	0.325508	0
0.96217	0.301966	0.399381	0.468742	1
0.432215	0.988033	0.067858	0.074884	0
0.096146	0.264949	0.124858	0.552386	0
0.69259	0.382476	0.323922	0.740827	1
0.24087	0.505574	0.499285	0.229094	1
0.830683	0.717169	0.967844	0.50378	0

```
Carbon: 0.40 mil
Nitrogen: 0.70 mil
Oxygen: 0.40 mil
Hydrogen: 0.60 mil
Data pushed successfull
```

Figure 4.11: Data Transmission between thing speak and Wi-Fi

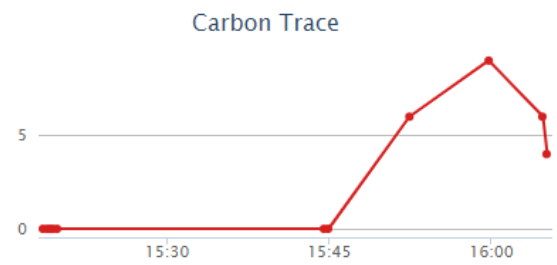


Figure 4.12 (a)

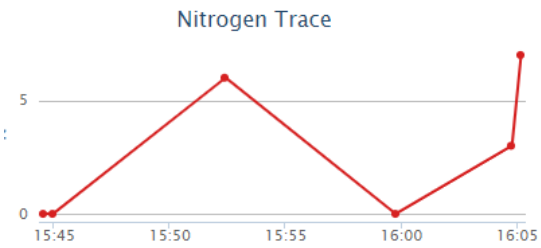


Figure 4.12 (b)

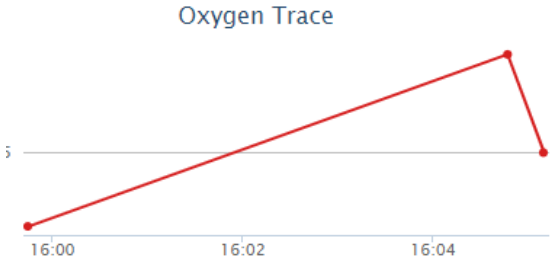


Figure 4.12 (c)

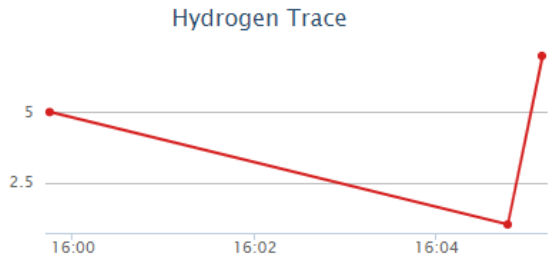


Figure 4.12 (d)

Figure 4.12: Graphic value for Explosive Traces Transmission

4.4 Discussion of the Results

This studies tend to develop an AI base system that can detect explosive trace automatically within an environment of interest using transfer learning model. The framework for the system was developed as shown in figure 3.1 base on deep transfer learning that can detect explosive trace with few explosive trace data with minimal time of training required. The designed framework is in such a way for the system to operate within a smart city by communicating with appropriate authority the presence of explosive trace to be able to take prompt action.

The explosive dataset obtained is a time serial dataset and was converted to a 2D dataset after normalization as presented in Figure 4.2 through serial data to image data generator as presented in section 3.3, this results is to generate an improved results for the CNN model.

The Gas-Net base model was development and with 70% of the data used for training the model, 20% for testing and 10% for validation for the 1,000 data points used, the system perform with high accuracy of 98.2% with an AUC of 1. Since the performance of the system high and acceptable as confirm by the AUC test, the knowledge of the model can be transferred for live test using fewer dataset based on deep transfer learning.

From the result of the experiment shown for the transfer learning model in Figure 4.8 through 4.10, it was discovered that less time is required to train transfer learning model against the CNN base

model. The transfer learning model converged faster with nearly zero losses for both training and validation experiment. Since the iteration of the first epoch has training accuracy of 99.7% and validation accuracy at 100%, this result confirms that the transfer learning model adjusted very quickly with the dataset to achieve very high and stable performance with the few data samples and small epoch size.

The result in figure 4.11 and 4.12 validate that deep transfer learning model will produce high performance metrics values after deploying different data on it with the ability to maintain such performance with reduced data. Deep transfer learning is appropriate for area base explosive trace detection where the system is expected to perform with high accuracy with less data and fast adaptation.

4.5 Benchmark of the results

The performance of the model is compared with other model such as Support Vector Machine, ImageNet, Random Forest and K-Near Neighbor, The Deep Transfer Learning for Explosive Trace Detection (DTLETD) outperformed all with an improve training accuracy of 99.7 and AUC of 0.89 as shown in Table 4.4. The detail performance is shown in table 4.5 and the graphical representation of accuracy and AUC shown in figure 4.13. The parameter setting use for the SVN, ImageNet, RNN, AlexNet shown in appendix H.

Table 4.4: Comparing the proposed model and other Machine Learning Models

MODEL	Validation Accuracy	Validation AUC	Training Time(s)
SVM	76	0.50	31.2
ImageNet	77	0.64	26.3
RNN	62%	0.63	28.8
AlexNet	67%	0.71	26.3
CNN	98.2	1.00	29.4
DTLETD	99.7	0.89	25.4

Table 4.5: Metrics of the Benchmarking Results

	ACC	PRECISION	RECALL	F-1 SCORE	SPECIFICITY	FPR	TPR
CNN	0.982	0.985	0.985	0.985	0.949	0.051	0.985
DTLETD	0.997	0.999	0.999	0.999	0.984	0.016	0.999
SVM	0.762	0.773	0.773	0.773	0.748	0.252	0.773
ImageNet	0.773	0.798	0.798	0.798	0.742	0.258	0.798
RNN	0.626	0.715	0.715	0.715	0.561	0.439	0.715
AlexNet	0.671	0.699	0.699	0.699	0.640	0.360	0.699

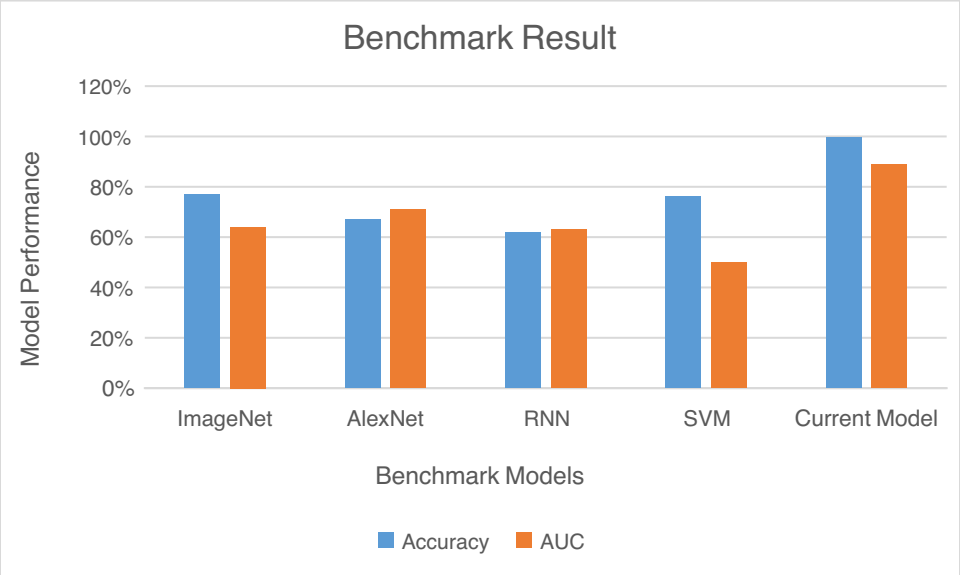


Figure 4.13: Comparing accuracy and AUC of Current Model with other models

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary

Attacks on people and several public organization has made explosive trace detection a concern on how best to secure sensitive environment of interest against potential attracts. The limitation of human apparatus with the introduction of AI models have led to the development of smart systems that can detect explosive trace within an environment automatically leveraging of machine leaning approach. Several attempts have been made in developing machine model that can accurately detect explosive trace leveraging on WSN technology.

In this work, a framework for area based explosive detection was designed to accurately utilize deep transfer learning model to detect explosive trace. The deep transfer learning model was developed to solve the problem need for quick adaptation of model and also to be able to accurately function with little available data. The system was validated using 10% of the available data and was found to have high accuracy. This result shows that deep transfer learning model can work well in detecting explosive trace very fast with little information available and that can be done even on edge devices.

5.2 Conclusion

Since terrorist attacks has become a global challenge in public places an AI system that utilizes WSN technology and deep transfer learning model is proposed. The system was able to detect explosive trace that compose of carbon, hydrogen, oxygen and Nitrogen component within an area.

The system recorded an accuracy of 98.2% with an AUC of 1 when the deep learning base model (CNN) was used. However, upon simulation, data were used to validate the transfer learning model, the model achieved an average prediction accuracy of 99.7%, with an average AUC value of about 0.89. The system also yielded a precession of 96% and recorded the least

training time compare with existing models. This result shows that the developed transfer learning model could still produce high performance metrics values after deploying different data on it. It shows that deep transfer learning model can adapt faster in detecting explosive trace and will work well on edge devices because of its ability to predict explosive traces well in the presence of few dataset.

5.3 Recommendations

This research focuses on detection of explosive trace that is only a particular type of explosive. The bulk type explosive is not considered, the two types of explosives can be detected by integrating different both chemical sensors and vision sensor through enhanced AI integrated model. This can bring about an enhance security of the area of interest against terrorist attack in the form of bombs.

5.4 Contributions to Knowledge

At the course of this research, we have provided the following contributions to knowledge:

i. A 2D gas data visualization for Deep learning model developed. The model generate explosive image data from explosive trace serial data. This is shown in section 3.5.2 it transform explosive trace serial data and produce image data that is suitable for the deep convolutional network. The accuracy of the model was increased when the image was generated. It also made the operation more robust.

ii. An explosive trace detection Framework was designed to show the stages of development in explosive trace detection. This design begins with the deep learning base model, to how layers of the models are been frozen to develop a new model. The framework is discussed in section 3.2 and figure 3.1 is the designed framework. The framework also show how real-time explosive data will be generated from sensor network to validate the model and anytime explosive trace is being detected, appropriated authority will be notified for prompt action. This has achieved the second objective of this research.

iii. An improved Explosive Trace Detection model (IETD) based on Deep transfer learning. A model for explosive trace detection was developed called DTLETD as

stated in objective three. Section 3.6.2 explained the procedure for the model, while development while figure 3.12 shows the model developmental diagram. The model was tested and recorded a better performance compare to existing model

iv. A light weighted model for explosive trace detection that can be run on edge computers. The DTLETD model developed is scalable and have a fast training rate that was able to perform optimally using limited data as recorded in the validation test in section 4.3.4 and proved with the graph in figure 4.9. This system is lighter than the normal deep learning model. This fulfilled objective three. The live deployment on edge device is beyond the scope of this work as discussed in the scope.

5.5 Future Research Directions

- One main recommendation for future direction is the development of ML model that can detect both explosive traces and bulk explosive the same time. The model should be able to detect explosive substance in different state.
- Considering WSN design and mapping that will comprehensibly cover the area of interest, this involves how the sensors should communicate effectively with one another and the base server.
- Deploying the proposed deep transfer learning model for real-time implementation on edge devices can be used by security agents to monitor sensitive areas of interest against explosive
- Another area to be considered is developing IOT base system that can communicate with security agent and with exact location of the explosive trace in an area, this will make the system part of smart city development.
- Another area that can be considered is sensor design that could lead in-cooperating ML algorithm that can improve the sensitivity of explosive trace detection.

REFERENCES

Adegoke, O., & Nic Daeid, N. (2021). Colorimetric optical nanosensors for trace explosive detection using metal nanoparticles: advances, pitfalls, and future perspective. *Emerging Topics in Life Sciences*, *5*(3), 367–379. https://doi.org/10.1042/ETLS20200281

Al-mousawi, A. J., & Al-mousawi, A. J. (2019). work and machine learning Magnetic Explosives Detection System (MEDS) based on wireless sensor network and machine learning . *Measurement*, 107112. https://doi.org/10.1016/j.measurement.2019.107112

AL-Mousawi, A. J., & K. AL-Hassani, H. (2018). A survey in wireless sensor network for explosives detection. *Computers and Electrical Engineering*, *72*, 682–701. https://doi.org/10.1016/j.compeleceng.2017.11.013

Aljojo, N., Chiroma, H., Mojeed, H., & Faruk, N. (2022). *A systematic review and Meta-data analysis on the applications of Deep A systematic review and Meta-data analysis on the applications of Deep Learning in Electrocardiogram*. *July*. https://doi.org/10.1007/s12652-022-03868-z

Almog, J., & Zitrin, S. (2009). Colorimetric Detection of Explosives. *Aspects of Explosives Detection*, 41–58. https://doi.org/10.1016/B978-0-12-374533-0.00004-0

Alom, Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Hasan, M., Essen, B. C. Van, Awwal, A. A. S., & Asari, V. K. (2019). *A State-of-the-Art Survey on Deep Learning Theory and Architectures*. 1–67. https://doi.org/10.3390/electronics8030292

Bajić, S. Ć. (2014). Analysis of the Possibility of Utalization of Honey Bees in Explosive Detection. *Polytechnic & Design*, *2*(1), 58–63. https://doi.org/10.19279/TVZ.PD.2014-2-1-06

Barrera, J. S., Echavarría, A., Madrigal, C., & Herrera-Ramirez, J. (2020). Classification of hyperspectral images of the interior of fruits and vegetables using a 2D convolutional neuronal network. *Journal of Physics: Conference Series*, *1547*(1). https://doi.org/10.1088/1742-6596/1547/1/012014

Barupal, D. K., & Fiehn, O. (2019). Generating the blood exposome database using a comprehensive text mining and database fusion approach. *Environmental Health Perspectives*, *127*(9), 2825–2830. https://doi.org/10.1289/EHP4713

Campbell, J. I. D., Fuchs-Lacelle, S., & Phenix, T. L. (2006). Identical elements model of arithmetic memory: Extension to addition and subtraction. *Memory and Cognition*, *34*(3),

633–647. https://doi.org/10.3758/BF03193585

Chowdhury, S. S., Tudu, B., Bandyopadhyay, R., & Bhattacharyya, N. (2008). Portable electronic nose system for aroma classification of black tea. *IEEE Region 10 Colloquium and 3rd International Conference on Industrial and Information Systems, ICIIS 2008*, 1–5. https://doi.org/10.1109/ICIINFS.2008.4798403

Chuen To, K., Ben-Jaber, S., & P. Parkin, I. (2020). Recent Developments in the Field of Explosive Trace Detection. *ACS Nano*, *14*(9), 10804–10833. https://doi.org/10.1021/acsnano.0c01579

Command, D., & Belvoir, F. (1978). U.s. army mobility equipment. *Security*.

Crocombe, R. A., Leary, P. E., & Kammrath, B. W. (2021). Portable Spectroscopy and Spectrometry. In *Portable Spectroscopy and Spectrometry*. https://doi.org/10.1002/9781119636489

Cunin, T., McGrath, S., & MacNamee, C. (2018). Environmental monitoring based on Internet of Things Technology. *Proceedings of the International Conference on Sensing Technology, ICST*, *2018-Decem*, 31–34. https://doi.org/10.1109/ICSensT.2018.8603658

Deming, R., Ilin, R., & Macintosh, S. (2017). *Deep Learning for Fusing Multi-Sensor Person-borne IED Data*.

Díaz-Ramírez, J. (2021). Machine Learning and Deep Learning. *Ingeniare*, *29*(2), 182–183. https://doi.org/10.4067/S0718-33052021000200180

Djedidi, O., Djeziri, M. A., Morati, N., Seguin, J., & Bendahan, M. (2021). Sensors and Actuators : B . Chemical Accurate detection and discrimination of pollutant gases using a temperature modulated MOX sensor combined with feature extraction and support vector classification. *Sensors and Actuators: B. Chemical*, *339*(November 2020), 129817. https://doi.org/10.1016/j.snb.2021.129817

Evans-Nguyen, K., Stelmack, A. R., Clowser, P. C., Holtz, J. M., & Mulligan, C. C. (2021). Fieldable Mass Spectrometry for Forensic Science, Homeland Security, and Defense Applications. *Mass Spectrometry Reviews*, *40*(5), 628–646. https://doi.org/10.1002/mas.21646

Fang, J., Hu, J., Wei, J., Liu, T., & Wang, B. (2020). An efficient resource allocation strategy for edge-computing based environmental monitoring system. *Sensors (Switzerland)*, *20*(21), 1–16. https://doi.org/10.3390/s20216125

Fang, J., & Ma, A. (2021). IoT Application Modules Placement and Dynamic Task Processing in Edge-Cloud Computing. *IEEE Internet of Things Journal*, *8*(16), 12771–12781. https://doi.org/10.1109/JIOT.2020.3007751

Fisher, D., R. Lukow, S., Berezutskiy, G., Gil, I., Levy, T., & Zeiri, Y. (2020). Machine Learning Improves Trace Explosive Selectivity: Application to Nitrate-Based Explosives. *The Journal of Physical Chemistry A*, *124*(46), 9656–9664. https://doi.org/10.1021/acs.jpca.0c05909

Fong, D. Y. (2017). Wireless sensor networks. *Internet of Things and Data Analytics Handbook*, 197–213. https://doi.org/10.1002/9781119173601.ch12

Frost, D. F. (1990). *Centralized Source of Information for the Military Working Dog Program*. 327. http://ez.library.latrobe.edu.au/login?url=https://search.proquest.com/docview/9709586?accountid=12001 http://ap01.alma.exlibrisgroup.com/view/uresolver/61LATROBE_INST/openurl?ctx_enc=info:ofi/enc:UTF-8&ctx_ver=Z39.88-2004&url_ctx_fmt=info:ofi/fmt:kev:mtx

Furton, K. G., & Myers, L. J. (2001). The scientific foundation and efficacy of the use of canines as chemical detectors for explosives. *Talanta*, *54*(3), 487–500. https://doi.org/10.1016/S0039-9140(00)00546-4

Gares, K. L., Hufziger, K. T., Bykov, S. V., & Asher, S. A. (2016). *Review of explosive detection methodologies and the emergence of standoff deep UV resonance Raman*. Journal of Raman Spectroscopy. https://doi.org/10.1002/jrs.4868

Gary, & Eiceman. (2006). working in an industrial or regulatory-based setting with limited appeal to academics. In *Ion Mobility Spectrometry* (2nd ed., pp. 5585–5588). CRC Press (an imprint of Taylor and Francis Group). https://doi.org/10.1021/ja0598560

Gazit, I., Lavner, Y., Bloch, G., Azulai, O., Goldblatt, A., & Terkel, J. (2003). A simple system for the remote detection and analysis of sniffing in explosives detection dogs. *Behavior Research Methods, Instruments, and Computers*, *35*(1), 82–89. https://doi.org/10.3758/BF03195499

Gill, P., Horgan, J., & Lovelace, J. (2011). Improvised explosive device: The problem of definition. *Studies in Conflict and Terrorism*, *34*(9), 732–748. https://doi.org/10.1080/1057610X.2011.594946

Girotti, S., Ghini, S., Maiolini, E., Bolelli, L., & Ferri, E. N. (2013). Trace analysis of pollutants by use of honeybees, immunoassays, and chemiluminescence detection. *Analytical and Bioanalytical Chemistry*, *405*(2–3), 555–571. https://doi.org/10.1007/s00216-012-6443-3

Gradišek, A., Midden, M. Van, Koterle, M., Prezelj, V., Strle, D., Kvasi, I., Zupaniˇ, E., Štefane, B., Brodnik, H., Trifkoviˇ, M., & Muševiˇ, I. (2019). Improving the Chemical Selectivity of an Electronic. *Sensors*, *19*(23), 1–15. https://doi.org/10.3390/s19235207

Haji, S. H., & Sallow, A. B. (2021). *IoT for Smart Environment Monitoring Based on Python : A Review*. *9*(1), 57–70. https://doi.org/10.9734/AJRCOS/2021/v9i130215

Hao, R., Zhao, J., Liu, J., You, H., & Fang, J. (2022). Remote Raman Detection of Trace Explosives by Laser Beam Focusing and Plasmonic Spray Enhancement Methods. *Analytical Chemistry*, *94*(32), 11230–11237. https://doi.org/10.1021/acs.analchem.2c01732

Hossny, Karim; Magdi, Salma; Hossny, Ahmad; Soliman, A. (2020). Data for: Detecting Explosives by PGNAA using KNN Regressors and Decision Tree Classifier". *Mendeley Data*, *V1*. https://doi.org/10.17632/y2k7b6mzmc.1

Iman, M., & Arabnia, H. (2022). *A Review of Deep Transfer Learning and Recent Advancements*. *January*. https://doi.org/10.13140/RG.2.2.15400.08963

J. Rodacy, P., F. A. Bender, S., J. Bromenshenk, J., B. Henderson, C., & Bender, G. (2002). The training and development of honeybees to detect explosives and other agentsof harm. *Proc. SPIE*, *4742*(2002), 474–481.

Jimenez, A. M., & Navas, M. J. (2007). Detection of explosives by chemiluminescence. *Counterterrorist Detection Techniques of Explosives*, 1–39. https://doi.org/10.1016/B978-044452204-7/50020-1

Jordan, M. I., & Mitchell, T. M. (2015). *Machine learning: Trends, perspectives, and prospects*. *349*(6245).

Junaid, H. M., Waseem, M. T., Khan, Z. A., Gul, H., Yu, C., Shaikh, A. J., & Shahzad, S. A. (2022). Fluorescent and colorimetric sensors for selective detection of TNT and TNP explosives in aqueous medium through fluorescence emission enhancement mechanism. *Journal of Photochemistry and Photobiology A: Chemistry*, *428*(October 2021), 113865. https://doi.org/10.1016/j.jphotochem.2022.113865

Kapitanova, K., Son, S. H., & Kang, K. D. (2010). Event detection in wireless sensor networks - Can fuzzy values be accurate? *Lecture Notes of the Institute for Computer Sciences, Social-*

Informatics and Telecommunications Engineering, *49 LNICST*, 168–184. https://doi.org/10.1007/978-3-642-17994-5_12

Kishore, K. R. V. K. K. V., Manthru, N. G., & Gudipati. (2019). Wireless nano senor Network (WNSN) for trace detection of explosives: The case of RDX and TNT. *Instrumentation Mesure Metrologie*, *18*(2), 153–158. https://doi.org/10.18280/i2m.180209

Kishore Kumar, R. V., & Murali, G. (2016). A survey on the present State-of-the-Art of explosives, detection methods and automatic explosive detection using wireless sensor network. *International Journal of Applied Engineering Research*, *11*(1), 504–510. https://doi.org/10.51301/vest.su.2021.i3.18

Kumar, R. V. K., Murali, G. B., & C, B. S. K. (2019). *An Accurate Methodology to Identify the Explosives using Wireless Sensor Networks*. *2003*, 2335–2347.

Li, H., Ota, K., & Dong, M. (2018). Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing. *IEEE Network*, *32*(1), 96–101. https://doi.org/10.1109/MNET.2018.1700202

Liu, R., Li, Z., Huang, Z., Li, K., & Lv, Y. (2019). Biosensors for explosives: State of art and future trends. *TrAC - Trends in Analytical Chemistry*, *118*, 123–137. https://doi.org/10.1016/j.trac.2019.05.034

López, P., Triviño, R., Calderón, D., & Guamán, A. V. (2017). Electronic nose prototype for explosive detection. *IEEE*, 3–6.

Mansoor, P. (2018). *improvised explosive device weapon*. https://www.britannica.com/technology/military-technology

Mao, J., Jiang, X., & Zhang, X. (2019). Analysis of node deployment in wireless sensor networks in warehouse environment monitoring systems. *Eurasip Journal on Wireless Communications and Networking*, *2019*(1), a. https://doi.org/10.1186/s13638-019-1615-x

Marshall, M., & Oxley, J. C. (2009). The Detection Problem. *Aspects of Explosives Detection*, 1–10. https://doi.org/10.1016/B978-0-12-374533-0.00001-5

Matin, M. A., & Islam, M. M. (2018). *Overview of Wireless Sensor Network Security Technology*. 3–24. https://doi.org/10.25236/iceeecs.2018.096

Minni, M., & Siddharth, S. (2016). *Border security robot*. *5*(2), 275–283. https://doi.org/10.5121/ijci.2016.5230

Mokalled, L., Al-husseini, M., Kabalan, K. Y., & El-hajj, A. (2014). Sensor Review for Trace

Detection of Explosives. *International Journal of Scientific & Engineering Research*, *5*(6), 337–350.

Mølgaard, L. L., Buus, O. T., Larsen, J., Babamoradi, H., Thygesen, I. L., Laustsen, M., Munk, J. K., Dossi, E., O'Keeffe, C., Lässig, L., Tatlow, S., Sandström, L., & Jakobsen, M. H. (2017). Improved detection of chemical substances from colorimetric sensor data using probabilistic machine learning. *Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XVIII*, *10183*, 1018307. https://doi.org/10.1117/12.2262468

Nguyen, Q. H., Ly, H. B., Ho, L. S., Al-Ansari, N., Van Le, H., Tran, V. Q., Prakash, I., & Pham, B. T. (2021). Influence of data splitting on performance of machine learning models in prediction of shear strength of soil. *Mathematical Problems in Engineering*, *2021*. https://doi.org/10.1155/2021/4832864

Obasi, C., Odaba, A., Ubadike, O., Ikharo, B., Ohemu, M. F., Oisamoje, V., & Chinedu, P. (2023). Augmented Security Framework using Internet of Things for Tracking Improvised Explosive Devices. *2023 2nd International Conference on Multidisciplinary Engineering and Applied Science (ICMEAS)*, *1*, 1–4. https://doi.org/10.1109/ICMEAS58693.2023.10379302

Omijeh, B., & Okemeka Machiavelli, A. (2019). Optimizing a Sensor to Detect Ammonium Nitrate Based IEDS in Vehicles Using Artificial Neural Networks. *American Journal of Neural Networks and Applications*, *5*(1), 1. https://doi.org/10.11648/j.ajnna.20190501.11

Pai, Peng; Xiaojin, Zhao;Xiaofang, Pan;Wenbin, Y. (2018). Gas Classification Using Deep Convolutional Neural Networks. *Sensors*, *18*(1), 1–6. https://doi.org/https://doi.org/10.3390/s18010157

Poling, A., Weetjens, B., Cox, C., Beyene, N. W., Bach, H., & Sully, A. (2011). Using Trained Pouched Rats To Detect Land Mines: Another Victory for Operant Conditioning. *Journal of Applied Behavior Analysis*, *44*(2), 351–355. https://doi.org/10.1901/jaba.2011.44-351

Regis, T., Cesar, L., Oliveira, M., & Chemical, A. (2018). Recent Development of Explosive Trace Detection. *American Chemical Society*, *10*(43), 1–60. https://doi.org/doi.org/10.1021/acsnano.0c01579

Rejeti, K., Murali, G., & Kumar, B. S. (2019). An Accurate Methodology to Identify the Explosives using Wireless Sensor Networks. *SSRN Electronic Journal*, *2003*, 2335–2347. https://doi.org/10.2139/ssrn.3362178

Royal Society of Chemistry. (2011). *Writing 'explosive' equations*. edu.rsc.org/uk-chemistry-olympiad/writing-explosive-equations-chemistry-olympiad-worked-answers

Sapir, G. I., & Giangrande, M. G. (2009). Explosives and Dangerous Chemical: Constitutional Aspects of Search and Seizure. In *Aspects of Explosives Detection* (First edit). Elsevier B.V. https://doi.org/10.1016/B978-0-12-374533-0.00012-X

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, *2*(3), 1–21. https://doi.org/10.1007/s42979-021-00592-x

Schmidhuber, J. (2014). Deep Learning in Neural Networks : An Overview. *Technical Report IDSIA*, *02*(14), 1–88.

Shahraki, H., Tabrizchi, M., & Farrokhpor, H. (2018). Detection of explosives using negative ion mobility spectrometry in air based on dopant-assisted thermal ionization. *Journal of Hazardous Materials*, *357*, 1–9. https://doi.org/10.1016/j.jhazmat.2018.05.054

Shrestha, Y. R., Krishna, V., & von Krogh, G. (2021). Augmenting organizational decision-making with deep learning algorithms: Principles, promises, and challenges. *Journal of Business Research*, *123*(January 2020), 588–603. https://doi.org/10.1016/j.jbusres.2020.09.068

Simi, S., & Ramesh, M. V. (2010). Real-time monitoring of explosives using wireless sensor networks. *Proceedings of the 1st Amrita ACM-W Celebration of Women in Computing in India, A2CWiC'10*. https://doi.org/10.1145/1858378.1858422

Simi, S., & Ramesh, M. V. (2011). *Wireless Sensor Network for Remote Detection of Explosives*. 60–71. https://doi.org/10.5220/0003116300600071

Smith, B. L., Boisdon, C., Young, I. S., Praneenararat, T., Vilaivan, T., & Maher, S. (2020). Flexible Drift Tube for High Resolution Ion Mobility Spectrometry (Flex-DT-IMS). *Analytical Chemistry*, *92*(13), 9104–9112. https://doi.org/10.1021/acs.analchem.0c01357

So, S., Sani, A. A., Zhong, L., Tittel, F., & Wysocki, G. (2009). Laser Spectroscopic Trace-Gas Sensor Networks for Atmospheric Monitoring Applications. *ESSA Workshop '09, April 16, 2009, San Francisco, California, USA*.

Song, K., Wang, Q., Liu, Q., Zhang, H., & Cheng, Y. (2011). A wireless electronic nose system using a Fe2O3 gas sensing array and least squares support vector regression. *Sensors*, *11*(1), 485–505. https://doi.org/10.3390/s110100485

Suganya, E., Sountharrajan, S., Shandilya, S. K., & Karthiga, M. (2019). Chapter 5 - IoT in

Agriculture Investigation on Plant Diseases and Nutrient Level Using Image Analysis Techniques. In *Internet of Things in Biomedical Engineering*. Elsevier Inc. https://doi.org/10.1016/B978-0-12-817356-5.00007-3

The National Research Council. (2004). Existing and Potential Standoff Explosives Detection Techniques. In *The national Academics Press*. https://doi.org/10.17226/10998

Thiesan, L., Hannum, D., Murray, D. W., & Parmeter, J. E. (2005). *Survey of Commercially Available Explosives Detection Technologies and Equipment 2004*.

Ul Hasan, N., Ejaz, N., Ejaz, W., & Kim, H. S. (2012). Malicious odor item identification using an electronic nose based on support vector machine classification. *1st IEEE Global Conference on Consumer Electronics 2012, GCCE 2012*, 399–400. https://doi.org/10.1109/GCCE.2012.6379638

Wang, W., Wang, C., Wang, Z., Yuan, M., Luo, X., Kurths, J., & Gao, Y. (2022). Abnormal detection technology of industrial control system based on transfer learning. *Applied Mathematics and Computation*, *412*(6), 821–832. https://doi.org/10.1016/j.amc.2021.126539

Wang, X., Zhang, H. R., & Zhang, C. J. (2005). Signals recognition of electronic nose based on support vector machines. *2005 International Conference on Machine Learning and Cybernetics, ICMLC 2005*, *August*, 3394–3398. https://doi.org/10.1109/icmlc.2005.1527528

Wasilewski, T., Gębicki, J., & Kamysz, W. (2021). Bio-inspired approaches for explosives detection. *TrAC - Trends in Analytical Chemistry*, *142*, 116330. https://doi.org/10.1016/j.trac.2021.116330

Wilkinson, A., Bevan, J., & Biddle, I. (2007). *Conventional Ammunition in Surplus*.

Wongwattanaporn, S. (2021). Machine Learning for Explosive Detection from Electronic Nose Datasets. *IEEE XPLORE*, 214–218. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9449327

Yaqoob, U., & Younis, M. I. (2021). Chemical gas sensors: Recent developments, challenges, and the potential of machine learning—a review. *Sensors*, *21*(8). https://doi.org/10.3390/s21082877

Yinon, J. (2007). Detection of explosives by mass spectrometry. *Counterterrorist Detection Techniques of Explosives*, 41–59. https://doi.org/10.1016/B978-044452204-7/50021-3

Zafar, F., Hameed, F., & Dar, M. A. (2017). *Detection COUNTRIES WITH THE HIGHEST NUMBER*. *7*(6), 126–134.

Zapata, F., & García-Ruiz, C. (2021). Chemical Classification of Explosives. *Critical Reviews in Analytical Chemistry*, *51*(7), 656–673. https://doi.org/10.1080/10408347.2020.1760783

Zheng, B. W. and C. (2023). Retracted: An Analysis of the Effectiveness of Machine Learning Theory in the Evaluation of Education and Teaching. *Wireless Communications and Mobile Computing*, *2023*, 1–1. https://doi.org/10.1155/2023/9839867

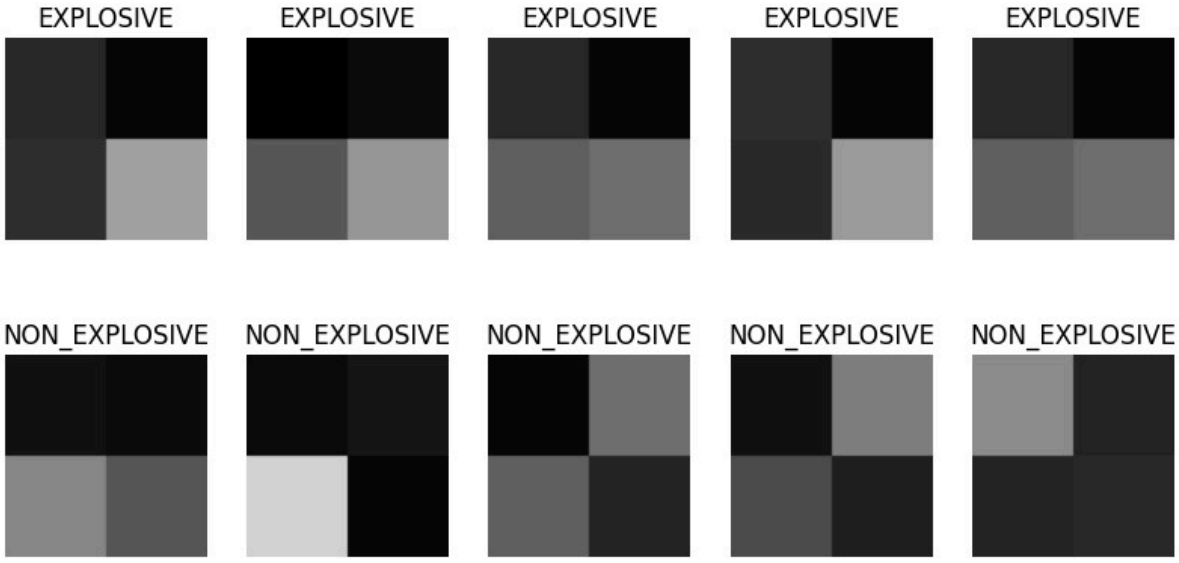
Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., & Member, S. (2020). *A Comprehensive Survey on Transfer Learning*. 1–31.

APPENDIX A: Program code for converting numerical data into 2D images

```
numeric_dataset = pd.read_csv('samples.csv')
image_width = 2
image_height = 2
min_val = numeric_dataset.min().min()
max_val = numeric_dataset.max().max()
numeric_dataset = (numeric_dataset - min_val) / (max_val - min_val)
class1_data = numeric_dataset[numeric_dataset['Target'] == 1]
class0_data = numeric_dataset[numeric_dataset['Target'] == 0]
class1_data = class1_data.drop('Target', axis=1)
class0_data = class0_data.drop('Target', axis=1)
output_dir = 'explosive_dataset'
os.makedirs(output_dir, exist_ok=True)
for i, row in class1_data.iterrows():
    # Convert the row data to a NumPy array and reshape it into an image
    image_data = row.to_numpy().reshape(image_height, image_width)
    # Create a grayscale image
    plt.figure(figsize=(2, 2))  # Adjust the figure size as needed
    plt.axis('off')  # Turn off axis labels
    plt.imshow(image_data, cmap='gray', vmin=0, vmax=1)  # Set cmap to 'gray' for grayscale
images
    # Save the image with a unique filename
    image_filename = os.path.join(output_dir, f'image_{i}.png')
    plt.savefig(image_filename, bbox_inches='tight', pad_inches=0, dpi=100)
    plt.close()
# Create a directory to save the generated images
output_dir = 'non_explosive_dataset'
os.makedirs(output_dir, exist_ok=True)
# Loop through each row in the numeric dataset and create an image for each
for i, row in class0_data.iterrows():
    # Convert the row data to a NumPy array and reshape it into an image
    image_data = row.to_numpy().reshape(image_height, image_width)

    # Create a grayscale image
    plt.figure(figsize=(2, 2))  # Adjust the figure size as needed
    plt.axis('off')  # Turn off axis labels
    plt.imshow(image_data, cmap='gray', vmin=0, vmax=1)  # Set cmap to 'gray' for grayscale
images
    # Save the image with a unique filename
    image_filename = os.path.join(output_dir, f'image_{i}.png')
    plt.savefig(image_filename, bbox_inches='tight', pad_inches=0, dpi=100)
    plt.close()
print("Image dataset created successfully.")
```

APPENDIX B: Sample 2D image of dataset



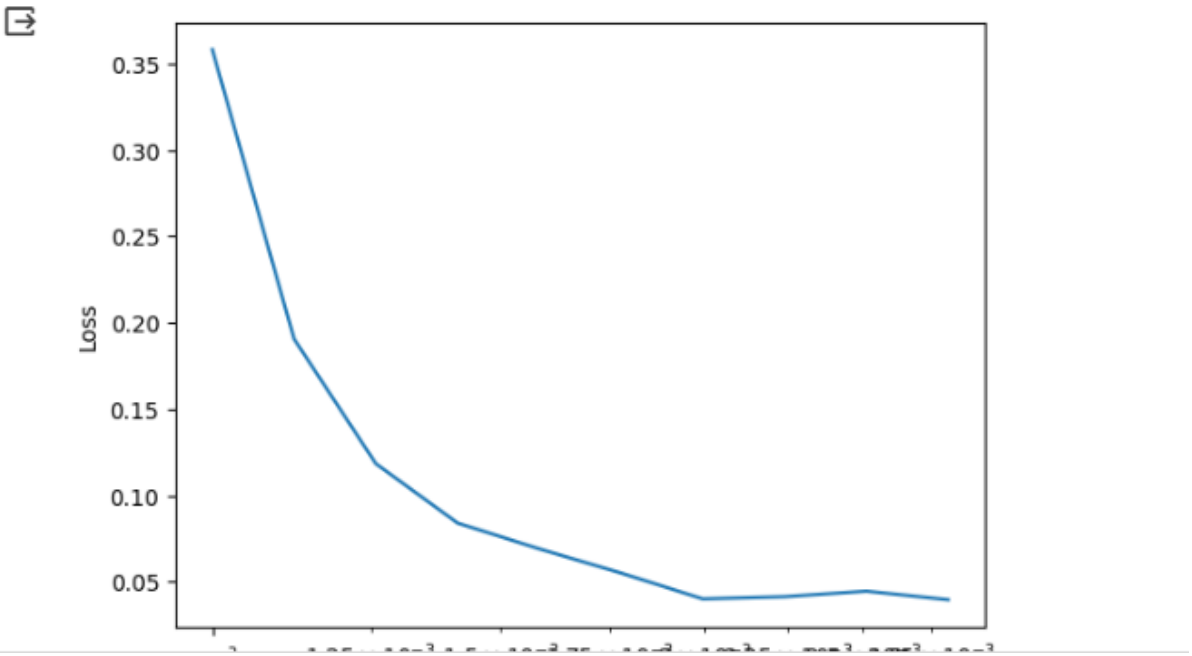
APPENDIX C; Transfer Learning Model Training for 10 Epochs

```
[ ] # Train the model
    history = model.fit(
        train_generator,
        steps_per_epoch=train_generator.samples // batch_size,
        epochs=epochs,
        validation_data=validation_generator,
        validation_steps=validation_generator.samples // batch_size,
        callbacks=[lr_scheduler]
    )
```

```
Epoch 1/10
218/218 [==============================] - 1014s 5s/step - loss: 0.3580 - accuracy: 0.8417 - val_loss: 0.1296 - val_accuracy: 0.9224 - lr: 0.0010
Epoch 2/10
218/218 [==============================] - 28s 126ms/step - loss: 0.1906 - accuracy: 0.9256 - val_loss: 0.1648 - val_accuracy: 0.9052 - lr: 0.0011
Epoch 3/10
218/218 [==============================] - 29s 131ms/step - loss: 0.1184 - accuracy: 0.9558 - val_loss: 0.0424 - val_accuracy: 0.9607 - lr: 0.0013
Epoch 4/10
218/218 [==============================] - 28s 128ms/step - loss: 0.0841 - accuracy: 0.9671 - val_loss: 0.0197 - val_accuracy: 0.9829 - lr: 0.0014
Epoch 5/10
218/218 [==============================] - 28s 127ms/step - loss: 0.0692 - accuracy: 0.9756 - val_loss: 0.1166 - val_accuracy: 0.9607 - lr: 0.0016
Epoch 6/10
218/218 [==============================] - 27s 124ms/step - loss: 0.0552 - accuracy: 0.9811 - val_loss: 0.0140 - val_accuracy: 1.0000 - lr: 0.0018
Epoch 7/10
218/218 [==============================] - 27s 125ms/step - loss: 0.0402 - accuracy: 0.9845 - val_loss: 0.0046 - val_accuracy: 1.0000 - lr: 0.0020
```

APPENDIX D: Convergence of learning rate of the Transfer Learning Model

```
#import numpy as np
lrs = 0.001 * (10 ** (np.arange(10) / 20))
plt.semilogx(lrs, history.history["loss"])
plt.xlabel('Learning Rate')
plt.ylabel('Loss')
plt.show()
```



APPENDIX E: Initializing the Libraries and the colab directories

```
import os
import glob
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.callbacks import LearningRateScheduler
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix, classification_report
import numpy as np
```

```
[ ] from google.colab import drive
    drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
[ ] path='/content/drive/MyDrive/Dataset/'
    print("Dataset path:  ")
    print(os.listdir(path))

    train_dir=path+'Train'
    test_dir=path+'Test'
    val_dir=path+'Val'
    print("Sample path:  ")
    print(val_dir)
```

DATA AUGMENTATION SECTION

```
# Data augmentation for the training set (optional but recommended)
train_datagen = ImageDataGenerator(
    rescale=1./255,      # Normalize pixel values to [0, 1]
    rotation_range=40,  # Randomly rotate images
    width_shift_range=0.2,  # Randomly shift images horizontally
    height_shift_range=0.2,  # Randomly shift images vertically
    shear_range=0.2,  # Shear transformations
    zoom_range=0.2,  # Randomly zoom images
    horizontal_flip=True,  # Randomly flip images horizontally
    fill_mode='nearest'  # Fill mode for pixel values outside the image
)
```

```
[ ] # Validation and test sets should not be augmented
    validation_datagen = ImageDataGenerator(rescale=1./255)
    test_datagen = ImageDataGenerator(rescale=1./255)
```

APPENDIX F: Arduino code for proteus simulation

```
#include <ArduinoJson.h>

//Gas Sensor Pins
#define MQ4 A1
#define MQ135 A2
#define MQ7 A3

void setup()
{
Serial.begin(9600); // opens serial port, sets data rate 9600 bps
}

void loop()
{
root["C"] = ppm_N;
root["N"] = ppm_N;
root["O"] = ppm_O;
root["H"] = ppm_H;


root.prettyPrintTo(Serial);
Serial.println("");

//Minimum delay required for ThingSpeak to update is 16 seconds
delay(16000);
}
```

Code for Thingspeak channel:

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <ArduinoJson.h>
#include <SoftwareSerial.h>
#include "ThingSpeak.h"

SoftwareSerial mySerial(5, 6);
WiFiClient client; // Creating WiFiClient Object

//ThingSpeak Channel's API Keys
unsigned long myChannelNumber = CHANNEL NUMBER;
const char * myWriteAPIKey = "API KEY";
```

```
//Add your WiFi credentials here
const char * WIFI_SSID = "SSID";
const char * WIFI_PASSWORD = "PASSWORD";

void setup() {
Serial.begin(9600);
mySerial.begin(9600);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("connecting");
while (WiFi.status() != WL_CONNECTED) {
Serial.print(".");
delay(100);
}
Serial.println();
Serial.print("connected: ");
Serial.println(WiFi.localIP());
ThingSpeak.begin(client);
}

void loop() {
// Check WiFi Status
while (mySerial.available())
{
const size_t capacity = JSON_OBJECT_SIZE(7) + 100;
DynamicJsonBuffer jsonBuffer(capacity);
JsonObject& root = jsonBuffer.parseObject(mySerial);
if (!root.success()) {
Serial.println("parseObject() failed");
return;
}

float C = root["C"];
float N = root["N"];
float O = root["O"];
float H = root["H"];


Serial.print(C, 5);  Serial.print(",");
Serial.print(N, 5);  Serial.print(",");
Serial.print(O, 5);  Serial.print(",");
Serial.print(H, 5);

//Sending Gas Data to ThingSpeak
ThingSpeak.setField(1, C);
ThingSpeak.setField(2, N);
ThingSpeak.setField(3, O);
ThingSpeak.setField(5, H);
```

```
ThingSpeak.writeFields(myChannelNumber,myWriteAPIKey);
}
}
```

APPENDIX G

Total number of samples used = 1000

AUC = Area Under the ROC curve

The ROC curve is a plot of the True Positive Rate (TPR) on the vertical axis, given as

$$TPR = \frac{TP}{TP + FN}$$

Against the False Positive Rate (FPR) on the horizontal axis, given as

$$FPR = \frac{FP}{TN + FP}$$

TP = True Positive (Number of times the model predicted positive outcomes correctly)

FP = False Positive (Number of times the model predicted positive outcomes wrongly)

TN = True Negative (Number of times the model predicted negative outcomes correctly)

FN = False Negative (Number of times the model predicted negative outcomes wrongly)

$$ImageNet\ Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

$$= \frac{520 + 245}{520 + 0 + 245 + 235} = \frac{765}{1000} = 0.77 = 77\ \%$$

$$ImageNet\ AUC = 0.64$$

Confusion Matrix of AlexNet			
		Predicted Class	
		Non-Explosive	Explosive
Actual Class	Non-Explosive	**TP**	**FN**
		430	335
	Explosive	**FP**	**TN**
		0	235

$$ImageNet\ Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

$$= \frac{430 + 235}{430 + 335 + 235 + 0} = \frac{665}{1000} = 0.67 = 67\ \%$$

$$ImageNet\ AUC = 0.71$$

Confusion Matrix of RNN			
		Predicted Class	
		Explosive	Non-Explosive
Actual Class	Explosive	**TP**	**FN**
		332	345
	Non-Explosive	**FP**	**TN**
		35	288

$$ImageNet\ Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

$$= \frac{332 + 288}{332 + 345 + 228 + 35} = \frac{665}{1000} = 0.62 = 62\ \%$$

$$ImageNet\ AUC = 0.63$$

Confusion Matrix of SVM			
		Predicted Class	
		Explosive	Non-Explosive
Actual Class	Explosive	TP	FN
		521	**245**
	Non-Explosive	FP	TN
		0	**234**

$$ImageNet\ Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

$$= \frac{521 + 234}{521 + 245 + 234 + 0} = \frac{755}{1000} = 0.76 = 76\ \%$$

$$ImageNet\ AUC = 0.50$$

Summary

Model	ImageNet	AlexNet	RNN	SVM	Current Model
Accuracy	77%	67%	62%	76%	99.7%
AUC	0.64	0.71	0.63	0.50	0.89

APPENDIX H

RNN:

Dense Layers: 1

Activation function: sigmoid

droupout: 0.5

learning rate: 0.01

optimizer: rmsprop

SVM:

kernel type: sigmoid

c parameter: 0.5

ImageNet:

Dense Layers: 2

Dense layer = 4096 units, activation function ReLu

Dense layer = 4096 units, activation function ReLu

Dense layer = 1000 units, activation function sigmoid

Optimizer: SGD (0.9)

learning rate: 0.01

Batch size: 128

drupout:0.5

AlexNet:

Dense Layers: 2

Dense layer = 4096 units, activation function ReLu

Dense layer = 4096 units, activation function ReLu

Dense layer = 10 units, activation function sigmoid

Optimizer: SGD (0.9)

learning rate: 0.01

Batch size: 128

drupout:0.5