



PRIFYSGOL CYMRU
Y Drindod Dewi Sant
UNIVERSITY OF WALES
Trinity Saint David
SWANSEA - ABERTAWE



School of
Applied Computing

PREDICTIVE ANALYTICS FOR THREAT INTELLIGENCE IN CYBERSECURITY LOG MANAGEMENT

by

Zhiqi Lu

Supervisor: Carlene Campbel

Project submitted as part of the requirements
for the award of BSc (Hons) Computer Networks

September 2023

Declaration of Originality

I, ZHIQI LU declare that I am the sole author of this Project; that all references cited have been consulted; that I have conducted all work of which this is a record, and that the finished work lies within the prescribed word limits.

This work has not previously been accepted as part of any other degree submission.

Signed : zhiqi lu

Date : Sep 15, 2023

FORM OF CONSENT

I, ZHIQI LU hereby consent that my Project, submitted in candidature for the B.Sc.(Hons) Computer Networks degree, if successful, may be made available for inter-library loan or photocopying (subject to the law of copyright), and that the title and abstract may be made available to outside organisations.

Signed : zhiqi lu

Date : Sep 15, 2023

Copyright Acknowledgement

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to University of Wales Trinity Saint David, Swansea.

ABSTRACT/SYNOPSIS

The rapidly evolving landscape of cybersecurity demands innovative methods to detect, predict, and respond to threats. With the inundation of data logs from network activities, there is an imperative need to extract meaningful insights and enhance threat intelligence. This thesis focuses on the utilization of a hybrid deep learning model that combines Convolutional Neural Networks (CNN) for spatial feature extraction and Long Short-Term Memory (LSTM) networks for temporal sequence prediction. Further, the incorporation of the self-attention mechanism ensures the prioritization of more significant data segments, optimizing intrusion detection accuracy.

Two prominent datasets in the realm of cybersecurity, NSL-KDD and UNSW-NB15, serve as the foundation for training and evaluating the proposed model. The research underscores the potential of this approach by presenting commendable results: In a binary class scenario, the model achieved an accuracy of approximately 90.37%, a detection rate of 54.75%, a false positive rate of 8.26%, and an F1 score of 55.07%. Meanwhile, in the multi-class scenario, the validation accuracy reached approximately 89.89% with a false positive rate of 8.52%.

The structure of the thesis provides a comprehensive overview of the evolution of cybersecurity practices, traditional methods of network intrusion detection, the significance of log management, and the emergence of deep learning in cybersecurity. The detailed design and implementation of the neural network model are presented, emphasizing its application in real-world scenarios.

By advancing the analytical and predictive capabilities in network security log management, this research offers a significant contribution to cybersecurity literature. It not only challenges the conventional methods of intrusion detection but also offers insights for further enhancements in the domain.

TABLE OF CONTENT

Abstract/Synopsis	
Table of Content	
List of Figures	
List of Tables	
Acknowledgements	
Copyright Acknowledgement	ii
CHAPTER 1 INTRODUCTION	10
1.1 Introduction	10
1.2 Background	10
1.3 Aim	12
1.4 Objectives	12
1.5 Structure Of The Project	13
1.6 conclusion	15
CHAPTER 2 - RESEARCH & REVIEW OF LITERATURE.....	16
2.1 Introduction	16
2.2 Evolution of Cybersecurity Practices.....	16
2.3 Traditional Network Intrusion Detection and Log Management	18
2.3.1 Origins and Functionality of Traditional NIDS	18
2.3.2 Significance of Network Security Logging	19
2.3.3 Traditional Log Management and Analysis Techniques	20
2.4 Deep Learning's Emergence in Cybersecurity	21
2.5 Previous Works on Feature Extraction in Cybersecurity.....	22
2.6 Studies on Sequence Prediction in Log Management	24
2.7 The Attention Paradigm in Cybersecurity.....	25
2.8 Popular Datasets in Cybersecurity Research	26
2.9 Current Challenges in Cybersecurity Predictive Analytics	27
2.10 Conclusion	29
CHAPTER 3 - RESEARCH METHODOLOGY	30
3.1 Introduction	30
3.2 Research Overview	30
2.2.1 Research Process	32

2.2.2	Research Philosophy	32
2.2.3	Research Methodologies	33
3.2.4	Approach	35
3.2.5	Strategies.....	36
3.2.6	Choices	37
3.2.7	Time Horizons	38
3.2.8	Techniques And Procedures	38
3.3	Conclusion	39
CHAPTER 4 - DESIGN & IMPLEMENTATION.....		41
4.1	Introduction	41
4.2	problem formulation	41
4.3	Design.....	43
4.3.1	The Overall Architecture	43
4.3.2	Spatial Feature Analysis with CNN	44
4.3.3	Temporal Feature Analysis with LSTM	46
4.3.4	Integration of Self-Attention Mechanism.....	48
4.3.5	Hidden Layer Design	50
4.3.5	Combination Blocks	52
4.4	Implementation.....	53
4.4.1	Data Preprocessing	53
4.4.2	Building the Neural Network Model	57
4.4.3	Training and Optimization.....	59
4.4.4	Validation	61
4.4	Deployment Environment.....	62
4.5	Conclusion	64
CHAPTER 5 – TESTING AND EVALUATION.....		65
5.1	Introduction.....	65
5.2	Performance Metrics	65
5.3	Performance Results.....	67
5.3.1	Multiclass Results	67
5.3.2	Binaryclass Results.....	70
5.4	Conclusion.....	72
CHAPTER 6 - CONCLUSIONS AND RECOMMENDATIONS		73

6.1	Summary	73
6.2	Conclusion	74
6.3	Recoomendations for Further Research	75
CHAPTER 7 – REFLECTIONS		77
CHAPTER 8 - REFERENCES		79
APPENDICES		85
PROJECT MANAGEMENT		85
LOGBOOK		88
ETHICS FORM		88

LIST OF FIGURES

Figure 1: Research Process..... 32

Figure 2: The Architecture of Model 44

Figure 3: Histograms for numerical columns 55

Figure 4: Correlation Heatmap for Numerical Columns..... 57

LIST OF TABLES

ACKNOWLEDGEMENTS

First and foremost, I would like to express my profound gratitude to my supervisor, Carlene, for her unwavering guidance, patience, and invaluable insights throughout the journey of this thesis. Her expertise and encouragement were instrumental in shaping this work.

I would also like to extend my heartfelt appreciation to Nitheesh Kaliyamurthy and other esteemed professors at the Wales Institute of Science and Art, University of Wales Trinity Saint David, for their constructive feedback, continual support, and enlightening discussions that have enriched my research experience.

I owe a special debt of gratitude to my parents, Xiangjun Lu and Xiuyun Yu, whose moral encouragement coupled with material reward has been a driving force behind my perseverance and determination. Their unwavering belief in my capabilities and their sacrifices for my education have been my guiding light.

In the course of drafting and revising this manuscript, several tools played a crucial role in bridging the linguistic gap. I wish to acknowledge the utility of Grammarly in refining my prose, DeepL for assisting with some vital translations, and Scispace for polishing the articles, among others. These platforms were pivotal in ensuring the linguistic precision of this thesis.

Lastly, I wish to express my appreciation to my peers, friends, and everyone else who played a part, however small, in this academic journey. Your contributions, both tangible and intangible, have left an indelible mark on this work.

CHAPTER 1 INTRODUCTION

1.1 INTRODUCTION

This introductory chapter endeavours to establish the context and framework for the forthcoming exploration. It begins by painting the background, drawing upon the historical significance and current dynamics of cybersecurity and the emergent role of deep learning. From there, it clearly defines the aim, anchoring the direction of this research, and then breaks down the objectives to provide a granular understanding of the research milestones. The structure of the project is subsequently laid out, offering readers a roadmap to navigate through the chapters and understand the logical progression of the study. Through this chapter, a foundation is laid, setting the stage for a comprehensive delve into the intersections of predictive analytics, deep learning, and cybersecurity log management.

1.2 BACKGROUND

In the digital era, the importance of cybersecurity cannot be understated. With the rise of cyber threats and their potentially devastating consequences, the need for proactive security measures has become paramount. Threat intelligence, as a core component of cybersecurity, serves as an early warning system, detecting potential threats before they become full-blown attacks.

As we delve into the intricacies of cybersecurity, we must acknowledge the evolution of log management. Historically, managing logs was a straightforward process. However, with the proliferation of devices and the complexity of modern networks, traditional log management faces challenges, including dealing with high volumes of logs and variable formats. Predictive analytics emerges as a solution, offering a way to proactively detect and respond to threats by analyzing patterns and predicting future threats.

Yet, the landscape of predictive analytics in cybersecurity is undergoing a transformative change with the incorporation of deep learning. Deep learning, a subset of machine learning, has begun to play an increasingly pivotal role in cybersecurity. While the application of such advanced algorithms offers numerous

benefits, it also presents a set of unique challenges. One of the key advantages of deep learning is its ability to revolutionize threat intelligence and log analysis, enabling professionals to decipher patterns that would otherwise go unnoticed.

In addition to the overarching framework of deep learning, specific architectures like Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) Networks have demonstrated significant potential. CNNs, renowned for their prowess in feature extraction, have been foundational in numerous applications across various domains. On the other hand, LSTMs, with their emphasis on sequence prediction, can capture temporal patterns, making them invaluable for tasks where order and sequence matter.

Not only that, but the advent of attention mechanisms in neural networks has further enriched the landscape. The basic premise of attention mechanisms lies in their ability to handle sequences more effectively. Particularly, the self-attention model stands out. Its capability to dynamically generate weights for connections offers unprecedented advantages, especially in log management where the sequence and context are crucial.

To evaluate and refine these sophisticated models, datasets like UNSW-NB15 have gained prominence in the cybersecurity community. These datasets, given their relevance and depth, have become benchmarks in the field, facilitating research and validation of novel approaches. Significant research has been conducted using these datasets, providing a foundation upon which further advancements can be built.

However, there exists a noticeable gap in current research. Traditional network intrusion detection methods, while effective to some extent, possess limitations that call for innovative approaches. This is where the novel approach of combining deep learning architectures and attention mechanisms can be seen not just as an iteration, but as a significant leap forward.

In conclusion, the significance of this research is multifaceted. By aiming to enhance the analytical and predictive performance in threat intelligence, it has the

potential to redefine practices in network security log management. Moreover, even if this approach diverges from traditional methods, its innovative nature positions it as a valuable contribution to the cybersecurity field.

1.3 AIM

The main aim of this report is

to employ predictive analytics, leveraging deep learning architectures combined with attention mechanisms, to enhance threat intelligence in cybersecurity log management.

By focusing on the effective interpretation and prediction of patterns in network activities using the UNSW-NB15 dataset, this research endeavours to provide a more robust and proactive approach to threat detection and response. Through this aim, we hope to bridge existing gaps and contribute innovatively to the realm of network security, diverging from traditional network intrusion detection methods.

1.4 OBJECTIVES

In alignment with the primary aim of enhancing threat intelligence in cybersecurity log management using predictive analytics, the following objectives are set out to guide this research:

- Conduct an in-depth review of the literature pertaining to the use of deep learning architectures, specifically CNN, LSTM, and attention mechanisms, in cybersecurity. This will provide context and ground the research in existing knowledge.
- Familiarize with the intricacies of the UNSW-NB15 dataset, understanding the patterns, challenges, and peculiarities that might impact the deep learning models.
- Design and develop a hybrid model that combines the strengths of CNN for feature extraction, LSTM for sequence prediction, and the self-attention mechanism for effective weight generation.

- Utilize the aforementioned datasets to train the developed model. Ensure that appropriate validation techniques are applied to avoid overfitting and to ensure the model's robustness.
- Once the model is trained, evaluate its predictive performance in detecting potential threats. Compare its efficiency and accuracy with traditional methods to establish its superiority or areas of improvement.
- Dive deeper into the attention mechanisms to understand which features or sequences the model deems most important in threat prediction, offering insights into potential areas of focus in real-world applications.
- Based on initial results, fine-tune the model to optimize its performance further. This might include adjusting parameters, retraining certain layers, or incorporating feedback loops.
- Ensure that the entire research, especially the model development and evaluation stages, adheres to the highest standards and best practices in both deep learning and cybersecurity domains.
- Analyze the outcomes, emphasizing the model's successes, challenges faced, and lessons learned. This will form the basis for conclusions and recommendations.

These above objectives are set to be Specific, Measurable, Achievable, Relevant, and Time-bound (SMART) , which is crucial to the success and validity of this research.

1.5 STRUCTURE OF THE PROJECT

The structure of this thesis can be conceptualized in three primary segments. Initially, we lay the foundation through theoretical insights that guide the project's trajectory. This segment is anchored in a rigorous review of literature pertaining to predictive analytics, deep learning architectures, and their applications in cybersecurity log management. Such a comprehensive review not only broadens the knowledge spectrum but is indispensable for informed decision-making throughout the project.

Subsequently, the core of the project emerges in the second segment. This encompasses the formulation of the research problem, articulating the design, and implementing the proposed hybrid model. For training and validation, datasets, namely UNSW-NB15, will be leveraged. This part also delves into the technological infrastructure, detailing both software and hardware prerequisites integral to the project's success.

The terminal segment is a synthesis of our findings. It presents a detailed analysis based on the predictive performance of our model and contrasts these results against existing traditional methods. Beyond mere findings, this part encapsulates conclusions drawn, weaving together threads from previous chapters, and subsequently projects the pathway for potential future research and developments in the domain.

Chapter One offers an introduction, outlining the context and significance of the research, pinning down the aim and elucidating the objectives that steer the project.

Chapter Two delves deep into the expansive realm of existing literature, focusing on predictive analytics in cybersecurity, the evolution and applications of deep learning, and the transformative potential of attention mechanisms.

Chapter Three elucidates the research methodology, detailing the approach, criteria for dataset selection, and the rationale behind choosing specific deep learning architectures and mechanisms.

Chapter Four centers on the design and implementation of the hybrid model, detailing the integration of CNN, LSTM, and attention mechanisms, and the intricacies of training with the selected datasets.

Chapter Five is dedicated to testing and evaluation. It scrutinizes the model's performance metrics, examines its strengths and areas of improvement, and offers a comparative analysis with traditional methods.

Chapter Six draws conclusions and recommendations, summarizing insights garnered from the entire research journey and suggesting potential areas of exploration for future endeavors.

Chapter Seven provides reflections, offering a personal perspective on the research process, challenges faced, lessons learned, and the broader implications of the study.

1.6 CONCLUSION

This chapter served as the gateway to our research, laying the foundation by elucidating the backdrop against which our exploration is set. We initiated our discourse with a thorough understanding of the evolving cybersecurity landscape, the pressing need for predictive analytics, and the transformative potential of deep learning in this domain. By establishing a clear aim, we anchored the research's trajectory, ensuring a targeted approach throughout. The subsequent delineation of objectives provided a roadmap, offering milestones that guide the research methodology and outcomes. Furthermore, by mapping out the structure of the project, we ensured a logical and coherent progression through each phase of the study, from theoretical underpinnings to practical applications and reflections. As we transition to the subsequent chapters, this foundation will enable a deeper, informed engagement with the various facets of our exploration into predictive analytics for threat intelligence in cybersecurity log management.

CHAPTER 2 - RESEARCH & REVIEW OF LITERATURE

2.1 INTRODUCTION

In this chapter, we embark on an extensive journey through the evolution and intricacies of cybersecurity practices, understanding their roots, developments, and the current standing in the context of predictive analytics. Beginning with the historical evolution of cybersecurity mechanisms, we delve into the specifics of traditional network intrusion detection systems (NIDS) and log management, unearthing their origins, functions, and the critical role they play in the larger landscape of cybersecurity. As we transition from the traditional methodologies, the chapter explores the revolutionary impact of deep learning in cybersecurity, investigating its applications, successes, and limitations. An examination of feature extraction methods, sequence prediction in log management, and the role of attention mechanisms provides a comprehensive view of the current methodologies. Further enriching the understanding, we shed light on popular datasets pivotal to cybersecurity research, highlighting their significance, uses, and challenges. Concluding the chapter, we introspect the challenges faced by predictive analytics in cybersecurity, setting the stage for subsequent discussions on innovative solutions.

2.2 EVOLUTION OF CYBERSECURITY PRACTICES

Cybersecurity practices have evolved significantly over time, influenced predominantly by the escalating sophistication of cyber threats. This review seeks to traverse the historical journey of cybersecurity, charting the progression from rudimentary firewall systems to the cutting-edge intrusion detection systems and security log management of today.

Delving into its historical perspective, the narrative of cybersecurity finds its roots in the nascent stages of the internet and network security technologies during the 1970s and 1980s. During this era, the US Department of Defense played a seminal role by demarcating research and management parameters for networks, with an emphasis on bolstering military-grade network security (Fidler, 2017).

Additionally, examining from a policy standpoint, Renda (2022) intricately reviewed the institutional and normative strides in EU cybersecurity policy, shedding light on pivotal strategy papers disseminated by the European Commission. In a similar vein, Carrapico & Barrinha (2017) delved into the coherence of the EU as a fortified security actor, especially in the realm of cybersecurity. These scholarly pursuits not only offer invaluable insights into the policy-oriented evolution of cybersecurity but also anchor its practical implications.

Transitioning from these early doctrines, a conspicuous inflection point in the evolution of cybersecurity has been the conceptualization and maturation of firewall systems. As delineated by Khan et al. (2013), firewalls are quintessential sentinels, strategically positioned between internal and external networks. They meticulously filter network traffic, governed by a comprehensive set of security protocols. Initially, these systems were simplistic, primarily hinging on elementary packet filtering methodologies. Yet, as cyber threats grew in complexity, there was an undeniable push towards innovation. Firewalls evolved, not only embracing advanced features such as stateful inspection and application-level gateways but also integrating intrusion prevention systems (Khan et al., 2013).

In addition to firewall innovations, intrusion detection systems (IDS) have solidified their place in the cybersecurity evolution. Designed with the prime objective of detecting and neutralizing unauthorized infiltrations or malevolent activities within a network framework, initial IDS models leaned heavily on signature-based detection paradigms. This involved juxtaposing network traffic patterns against an established catalog of recognized attack signatures (Elsayed et al., 2015). Yet, as cyber malefactors employed increasingly intricate techniques, including the likes of polymorphic malware and elusive zero-day exploits, the traditional signature-based frameworks waned in efficacy. This paradigm shift beckoned the dawn of cutting-edge IDS stratagems, encompassing both anomaly detection and behavior-centric analysis (Elsayed et al., 2015).

Beyond IDS, the value of log data for security analysis has been emphasized, with tools such as big data analytics playing a pivotal role in decoding the intricate relationship between security and data events (Jeon et al., 2016). A deeper

literature review on the history of network security log management suggests that our understanding and application of these systems need to be far more comprehensive and dynamic. Anastopoulos & Katsikas (2019) posited that while many log management infrastructures are grounded in industry best practices, they often fail to adapt to the evolving threat landscape.

Furthermore, the proliferation of the Internet of Things (IoT) has heralded a new era, accentuating the pressing challenges of cybersecurity. With IoT-augmented cyber-physical infrastructures becoming lucrative targets for cyber onslaughts, there's an impending demand for revolutionary cybersecurity resolutions (He et al., 2016). In this vein, the amalgamation of evolutionary computation with other computational intelligence modalities emerges as a beacon of promise, offering tangible solutions to the burgeoning security conundrums in IoT-integrated systems (He et al., 2016).

2.3 TRADITIONAL NETWORK INTRUSION DETECTION AND LOG MANAGEMENT

2.3.1 Origins and Functionality of Traditional NIDS

In the ever-evolving domain of network security, Traditional Network Intrusion Detection Systems (NIDS) have historically been important in the surveillance and detection of unauthorized activities across computer networks. Originating from the foundational need to safeguard digital infrastructures, these systems have consistently proved indispensable in myriad settings, not least among which are the Wireless Sensor Networks (WSN) (Manjula, 2022).

One of the cardinal functionalities that define traditional NIDS is their traffic analysis prowess. As Wanda (2020) elucidates, this involves an astute monitoring of the ingress and egress network traffic, laying the groundwork for both intrusion detection and its subsequent prevention. Delving deeper into their capabilities, Zheng et al. (2013) highlight the innate proficiency of these setups to adeptly

identify and categorize a plethora of attacks, tracing them back to their origins and charting out the intricate web of malicious activities.

Yet, while their benefits are manifest, the design and implementation of traditional NIDS are not devoid of challenges. Karim et al. (2017) acknowledge their widespread adoption in network security, but this has not eclipsed the inherent limitations, particularly when juxtaposed against modern network paradigms characterized by burgeoning traffic volumes and intricate topologies. Colajanni et al. (2007) articulate these challenges succinctly, pointing to the centralized architecture inherent to traditional NIDS, which often grapples with the complexities of present-day networks. Adding another layer of intricacy is their sometimes pronounced impact on network performance, which Manjula (2022) astutely observes as a constraint, especially in handling multifaceted network traffic data.

Nevertheless, in the face of these challenges, the research community has been anything but complacent. Pioneering endeavors, such as the ones underscored by Zeng et al. (2019), have ventured into the realm of deep learning, positing its potential in traffic classification and intrusion detection. This proffers an innovative lens through which the future of NIDS can be envisaged—a confluence of traditional methodologies and cutting-edge techniques.

2.3.2 Significance of Network Security Logging

In the labyrinthine expanse of modern cybersecurity, network security logging emerges as a cardinal linchpin. As elucidated by Wu et al. (2017), this intricate process, far from being merely about data accumulation, delves deep into the meticulous collection and discerning analysis of logs, all geared towards bolstering network security. This facet underscores the importance of log management in cybersecurity, shaping it as the protective bedrock against a gamut of potential cyber onslaughts.

However, it's not only the collection that's pivotal; the depth of insight gleaned from log analysis has proven transformative, especially in the realm of threat detection. Ambre & Shekokar (2015) and Tuor (2017) collectively spotlight the role of log analysis in deciphering abnormal network activity, bringing the often-overlooked insider threats into sharp focus. By continuously monitoring these logs, it becomes feasible to weave a protective tapestry, tracing every nuance of activity and thereby shielding networks from potential security transgressions.

In addition to these facets, the multifaceted realm of log management, encompassing both log analysis and the correlative events that spring from it, illuminates the pathogenic roots of cyberattacks, affording networks the critical foresight to preemptively safeguard against breaches (Ambre & Shekokar, 2015). Notably, as Sarker et al. (2022) opine, the leveraging of clustering techniques within log data analysis presents a novel frontier in cybersecurity. By unraveling the latent patterns and intricate structures obscured within logs, clustering techniques bequeath to security practitioners invaluable tools for deciphering anomalies and deftly detecting cyber-attacks.

To encapsulate, Sadik et al. (2020) aptly underscore the sentiment, positing that network security logging, complemented by astute log analysis, remains an indispensable cornerstone in the construction of a robust cybersecurity ecosystem. The synergetic blend of log management and its subsequent analysis not only fortifies the digital landscape but also ensures its sustainable evolution against ever-emerging threats.

2.3.3 Traditional Log Management and Analysis Techniques

In the intricate domain of data management, traditional log management and analysis techniques stand as venerable pillars. These methods, as highlighted by Tang et al. (2011), have found applicability across a spectrum of fields, ranging from biology and conservation to computer science and forestry. The methodologies from yesteryears, deeply rooted in established practices, have

served as the primary tools to parse and understand the deluge of log data for generations.

Yet, as with all things of antiquity, these traditional techniques are not without their impediments. Diving deeper, one begins to see that the erstwhile paradigms of log analysis face significant hurdles, especially in the current age of burgeoning data. For instance, the task of extracting system events from the raw textual logs poses a notable challenge. Given the often succinct nature of log messages, coupled with their expansive vocabulary, traditional text clustering techniques struggle to be efficacious (Tang et al., 2011). Furthermore, the sheer volume and intricacy of logs in our contemporary IT services landscape magnify the task, making it evermore arduous to extract actionable insights for problem diagnosis (Tak et al., 2019).

Recognizing these limitations intrinsic to traditional methods, the world of academia has been fervently exploring innovative avenues. Solutions like Priolog, as proposed by Tak et al. (2019), try their best to reduce the log volume to a distilled subset of pivotal and germane entries. Moreover, pioneering strides have been taken in the realms of association rule mining and automated planning tailored for security event log analysis (Khan & Parkinson, 2018). Also worth noting is the SLOGERT framework, designed for the automatic construction of log knowledge graphs (Ekelhart et al., 2021). Such novel techniques seek not just to supplant the limitations of their traditional counterparts but also to elevate the whole discipline of log management and analysis to newfound zeniths.

2.4 DEEP LEARNING'S EMERGENCE IN CYBERSECURITY

Deep learning has revolutionized various sectors, including cybersecurity. Deep learning techniques, which are based on parallel processing and can handle large volumes of data, have proven to be practical and effective in addressing complex cybersecurity issues. These techniques have been applied in intrusion detection, attack detection, and malware detection.

One successful application of deep learning in cybersecurity is game-based learning for high school cybersecurity education. Game-based learning allows students to explore different aspects of cybersecurity in a learning context, improving their understanding and skills in this field (Jin et al., 2018).

Deep learning has also been used for the detection and classification of cyber-attacks in IoT communication networks. The development of intelligent and autonomous deep learning-based systems, such as the IoT-IDCS-CNN, has shown impressive performance in detecting and classifying cyber-attacks in IoT networks (Al-Haija & Zein-Sabatto, 2020).

In the context of cybersecurity, deep learning offers several advantages over traditional machine learning methods. Deep learning techniques, such as neural networks, have better performance, especially when dealing with large amounts of security datasets (Sarker, 2021). These techniques can effectively learn and extract meaningful features from complex and high-dimensional data, improving the accuracy and efficiency of cybersecurity systems (Berman et al., 2019).

However, the security of deep learning models in adversarial environments is still a challenge. Adversarial learning methods, such as generative adversarial networks, have raised questions about the security of deep learning models and the evaluation of their performance in the presence of adversarial examples (Peng et al., 2020).

To address the increasing need for cybersecurity and intrusion detection systems, machine learning and deep learning methods have been extensively studied and reviewed. These studies have examined various techniques, including deep autoencoders, restricted Boltzmann machines, recurrent neural networks, and generative adversarial networks, and their applications in cybersecurity (Berman et al., 2019).

2.5 PREVIOUS WORKS ON FEATURE EXTRACTION IN CYBERSECURITY

Previous works on feature extraction in cybersecurity have employed various methods. These methods have shown varying levels of success, each with their own advantages and shortcomings.

One approach to feature extraction in cybersecurity is the use of machine learning and deep learning techniques. For example, Rackevičienė et al. (2021) applied machine learning and deep learning approaches to extract cybersecurity terms from text. Their pilot study demonstrated high results in extracting cybersecurity terms even with limited resources. Similarly, Gasmi et al. (2019) used an LSTM approach for information extraction of cybersecurity concepts. They found that traditional statistical-based extraction methods, although achieving good results, relied heavily on feature engineering.

Another approach to feature extraction in cybersecurity is the use of neural networks. Topolski (2021) applied feature extraction methods for chemical risk classification in the pharmaceutical industry. They found that feature extraction improved the classification quality compared to models without feature extraction. Wang et al. (2020) developed an explainable machine learning framework for intrusion detection systems, which involved extracting important features from IDSs using neural networks.

In addition to machine learning and neural networks, other methods have been used for feature extraction in cybersecurity. For instance, Fatani et al. (2021) developed new feature extraction and selection methods for an IoT Intrusion Detection System using swarm intelligence algorithms. They utilized conventional neural networks for feature extraction and an alternative feature selection approach using the Aquila optimizer. Shen et al. (2020) proposed a distant supervised relation extraction model based on a deep residual convolutional neural network and attention mechanism for constructing a cybersecurity knowledge graph.

While these previous works have shown success in feature extraction for cybersecurity, they also have their limitations. For example, Gasmi et al. (2019) noted that off-the-shelf NLP tools were not capable of extracting security-related

entities and their relations. Li et al. (2006) highlighted that Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA), two popular linear dimensionality reduction methods, had their limitations in extracting the most discriminant features and stability, respectively.

2.6 STUDIES ON SEQUENCE PREDICTION IN LOG MANAGEMENT

Studies on sequence prediction in log management have explored various methods. One approach to sequence prediction in log management is the extraction of log templates from the original log sequence. Yang et al. (2019) studied the detection of anomalies in the log template sequence corresponding to the original log sequence. They proposed the nLSALog framework, which considers a log sequence as a series of events represented by log templates. This approach aims to detect anomalies in the log template sequence, providing insights into log management and security.

Another method for sequence prediction in log management is the use of log event vectors and log sequence vectors. Wang et al. (2020) proposed LogEvent2vec, a framework for anomaly detection in large-scale logs in the Internet of Things (IoT). They transformed the coordinates of words into log event vectors and log sequence vectors, enabling the detection of anomalies in log sequences. This approach utilizes distributional lexical-contrast embedding models to extract semantic information from log templates.

Traditional text clustering techniques have limitations when applied to log data due to the large vocabulary of log messages. Zheng et al. (2011) highlighted the challenge of generating system events from raw textual logs and the poor performance of traditional text clustering techniques. They proposed the LogSig method, which addresses these limitations and improves log analysis.

In recent years, LSTM models have gained popularity for sequence prediction tasks, including log management. Lv et al. (2021) proposed ConAnomaly, a log-based anomaly detection model that combines a log sequence encoder (log2vec) and a multi-layer LSTM network. This model utilizes the Word2vec model to

vectorize words in log content and obtain sequence vectors. The LSTM network is then used to detect anomalies in log sequences.

Furthermore, there have been efforts to improve the efficiency and accuracy of log pattern mining in system log messages. Harding et al. (2021) introduced Sequence-RTG (Sequence-Ready-To-Go), a more efficient and production-ready version of log pattern mining. This approach demonstrated improved performance in terms of speed and accuracy in data sets from the literature.

2.7 THE ATTENTION PARADIGM IN CYBERSECURITY

The attention mechanism, known primarily for its role in the Transformer architecture, occupies a distinct niche in the vast landscape of computational methods. At its core, this mechanism was designed to selectively focus on specific segments of data, eschewing irrelevant portions, thereby mirroring the human cognitive process of 'paying attention.' Introduced by Vaswani in 2017, the Transformer architecture, constructed exclusively on attention mechanisms, has primarily been the stalwart of natural language processing (NLP) tasks. The attention-driven approach has showcased unparalleled efficiency in discerning intricate relationships in data, especially in sequential tasks where temporal relationships are paramount.

However, a more in-depth examination into the nexus of attention mechanisms and cybersecurity reveals a landscape that is somewhat barren. Although Vaswani's work did not venture into the precincts of cybersecurity, subsequent adaptations of the Transformer model began to make inroads into this domain. An illustrative example can be found in the work by Zhang et al. (2022). They embraced the Transformer's attention mechanism for a cybersecurity-specific NLP task—named entity recognition. This application stands testament to the mechanism's potential in the cybersecurity realm, illuminating its prowess in capturing pertinent information and unraveling underlying patterns intrinsic to cybersecurity data.

Yet, what stands out conspicuously is the relative scarcity of works that exploit the attention mechanism within cybersecurity. The few endeavors that exist seem concentrated within niche intersections of NLP and cybersecurity, leaving vast tracts of cybersecurity untouched by the attention paradigm. This observed gap becomes even more intriguing given the complexity and dynamism that characterizes cybersecurity data, making it a domain ripe for methods that can focus on relevant portions while sidelining noise and redundancies.

In conclusion, while the attention mechanism, particularly as encapsulated within the Transformer architecture, has transformed several computational domains, its foray into cybersecurity appears somewhat tentative and circumscribed. This landscape provides a fertile ground for innovation, offering researchers a chance to harness this potent mechanism to untangle the intricate web of challenges in cybersecurity. As part of the ongoing literature review, this paucity of attention-based models in cybersecurity establishes a compelling backdrop, setting the stage for a more detailed exploration of the model in the subsequent methodology section.

2.8 POPULAR DATASETS IN CYBERSECURITY RESEARCH

Popular datasets in cybersecurity research play a crucial role in advancing the field by providing researchers with real-world data to develop and evaluate cybersecurity solutions. Two widely recognized datasets in the cybersecurity community are the UNSW-NB15 dataset and the NSL-KDD dataset.

The UNSW-NB15 dataset is another widely used dataset in cybersecurity research (Moustafa & Slay, 2015). The dataset contains network traffic data collected from a real-world environment, including both normal and malicious traffic. Researchers have used the UNSW-NB15 dataset to develop and evaluate intrusion detection systems. For example, compared the performance of different machine learning models using the UNSW-NB15 dataset Shaukat et al. (2020).

The NSL-KDD dataset is a benchmark dataset for intrusion detection systems (IDSs) (Ghazal & Mjlae, 2022). It was developed as an improvement over the

original KDD Cup '99 dataset, which was criticized for not accurately representing real-world traffic (Protic, 2018). Researchers have used this dataset to develop and evaluate machine learning techniques for detecting network attacks. For example, Anwer et al. (2021) used the NSL-KDD dataset to detect malicious network traffic using three machine learning techniques. Although, NSL-KDD is not used in this paper. It will be utilized to test our model for future improvement.

Of course, these two datasets are just the tip of the iceberg. Ferrag et al. (2020) present a survey of deep learning approaches for cybersecurity intrusion detection and classify 35 well-known cyber datasets into seven categories. These categories include network traffic-based datasets, electrical network-based datasets, internet traffic-based datasets and so on. Chua & Salam (202) mention the CIC-IDS2017 dataset as one of the most popular datasets in recent years. This dataset was created by the Canadian Institute for Cybersecurity (CIC) in 2017 and is widely used for network-based intrusion detection research. Ma et al. (2021) demonstrate the effectiveness of their proposed model for cybersecurity named entity recognition using an open-source dataset containing an office security bulletin, security blogs, and the Common Vulnerabilities and Exposures list. Trong et al. (2020) discuss the development of a cybersecurity-focused dataset for event detection (CASIE) to aid research in this domain. They highlight the need for domain-specific datasets to capture the divergences in lexical forms, sentence structures, and domain expertise in the cybersecurity domain.

2.9 CURRENT CHALLENGES IN CYBERSECURITY PREDICTIVE ANALYTICS

Cybersecurity predictive analytics plays a crucial role in identifying and mitigating potential cyber threats. However, there are several limitations and challenges that need to be addressed to enhance the effectiveness of these techniques.

One of the limitations in current predictive analytics techniques is the communication gap within cybersecurity workforce teams (Kour & Karim, 2020). Effective communication of cyber-threat indicators is essential for the timely identification and response to potential threats. The literature emphasizes the

importance of having the right personnel and teams in place to facilitate this communication (Kour & Karim, 2020). Addressing this limitation requires improving the coordination and collaboration among cybersecurity professionals to ensure the effective sharing of threat intelligence.

Another challenge in cybersecurity predictive analytics is the need for advanced security analytics and automation to identify, respond to, and prevent security breaches (Kour et al., 2019). The current reactive nature of cybersecurity research and techniques leaves sensitive systems vulnerable to attacks due to unpatched systems and undetected exploits (Adams et al., 2013). To overcome this challenge, there is a need to develop proactive approaches that can anticipate and prevent cyber threats before they occur. This can be achieved through the integration of machine learning and advanced analytics techniques to analyze and interpret large volumes of data in real-time (Sarker et al., 2020).

Recency analysis is another issue in cybersecurity predictive analytics (Sarker et al., 2020). Recent malicious behavioral patterns are more likely to be significant in predicting unknown attacks compared to older ones. Therefore, effectively incorporating the concept of recency analysis into cybersecurity solutions is crucial for improving the accuracy and effectiveness of predictive analytics.

Furthermore, the complex nature of cybersecurity systems poses challenges in securing them against cyber disruptions (Young & Leveson, 2013). Traditional safety approaches have limitations that need to be addressed to enhance security. Adopting a systems thinking approach to safety can help in addressing these challenges and securing complex systems against cyber disruptions (Young & Leveson, 2013).

Additionally, the lack of available data on cyber risk poses a serious problem for stakeholders in tackling cybersecurity challenges (Cremer et al., 2022). The scarcity of quantitative assessment regarding the cognitive aptitudes, work roles, or team organization required by cybersecurity professionals hinders the development of effective cybersecurity strategies (Stevenson & Thomson, 2018). To overcome this challenge, there is a need for open databases and

comprehensive data collection efforts to support research and decision-making in cybersecurity.

2.10 CONCLUSION

This chapter provided a comprehensive overview of the cybersecurity landscape, tracing its journey from rudimentary practices to sophisticated predictive analytics. We discerned the progression from basic firewalls to advanced NIDS and the indispensable role of log management. Deep learning's transformative role in cybersecurity was presented, with specific focus on its applications and challenges. By evaluating earlier methods of feature extraction and sequence prediction, the chapter illustrated the continuous evolution of methodologies, culminating in the attention paradigm's contribution. Highlighting the significance of datasets such as NSL-KDD and UNSW-NB15, we delved into the invaluable research they have facilitated and the criticisms they face. Finally, we reflected on the limitations and challenges in the existing predictive analytics techniques, underlining the pressing need for innovative solutions. Through this journey, the chapter served as a foundation for understanding the complex tapestry of cybersecurity research and practices.

CHAPTER 3 - RESEARCH METHODOLOGY

3.1 INTRODUCTION

In this chapter, we embark on a comprehensive journey that delves deep into the methodologies, approaches, and decisions shaping this research on Predictive Analytics in Cybersecurity. Beginning with a clarified purpose and the significance of our research design, we magnify the role of machine learning, especially its symbiosis with cybersecurity threat intelligence. Each subsequent section serves as a scaffold, meticulously building upon the previous, offering clarity and direction. From discussing the philosophical foundations of our work to the specific datasets, methods, and strategies, this chapter presents a panoramic view of our research process. It delineates the rationale behind every choice, ensuring the reader grasps not just the 'what' but the 'why' of our scientific endeavor.

3.2 RESEARCH OVERVIEW

The domain of cybersecurity is in perpetual evolution, often catalyzed by the dynamic nature of threats and the need for steadfast defense mechanisms. Within this milieu, the current research serves a twofold purpose. Firstly, it aims to provide an overarching perspective on the interdisciplinary juncture of machine learning and cybersecurity. The nexus between these domains has proven to be fertile ground for innovation, primarily because of machine learning's prowess in deciphering patterns and predicting outcomes based on vast datasets. In addition, this research is poised to delineate the nuances of the chosen research design, thereby shedding light on its pertinence and the anticipated contributions to the realm of cybersecurity.

The research design has been meticulously crafted, keeping in mind the intricacies and the multifaceted nature of cybersecurity challenges. This structure draws heavily from the groundbreaking work of Creswell (2014), who emphasized the importance of a convergent design in research. Creswell's paradigm, as elucidated in his seminal paper, champions the amalgamation of both qualitative and quantitative methodologies, fostering a comprehensive research approach.

Within the context of cybersecurity, a domain riddled with both observable threats and abstract patterns, the adoption of a mixed-methods approach becomes imperative. Empirical analysis, in this scheme, serves as the linchpin. Its emphasis underscores the belief that in cybersecurity, real-world, tangible results often hold precedence over mere theoretical constructs. The chosen research design, therefore, is not merely a procedural roadmap; it is a strategic choice. It not only nurtures rigorous empirical investigations by analyzing real-time data and observable patterns but also intertwines these findings with theoretical constructs to facilitate a deeper, holistic understanding of the cybersecurity landscape.

Figure 1 provides a step-by-step breakdown of the research process, from inception to conclusion, reflecting the strategic choices and methodological rigor encapsulated in the research design.

2.2.1 Research Process

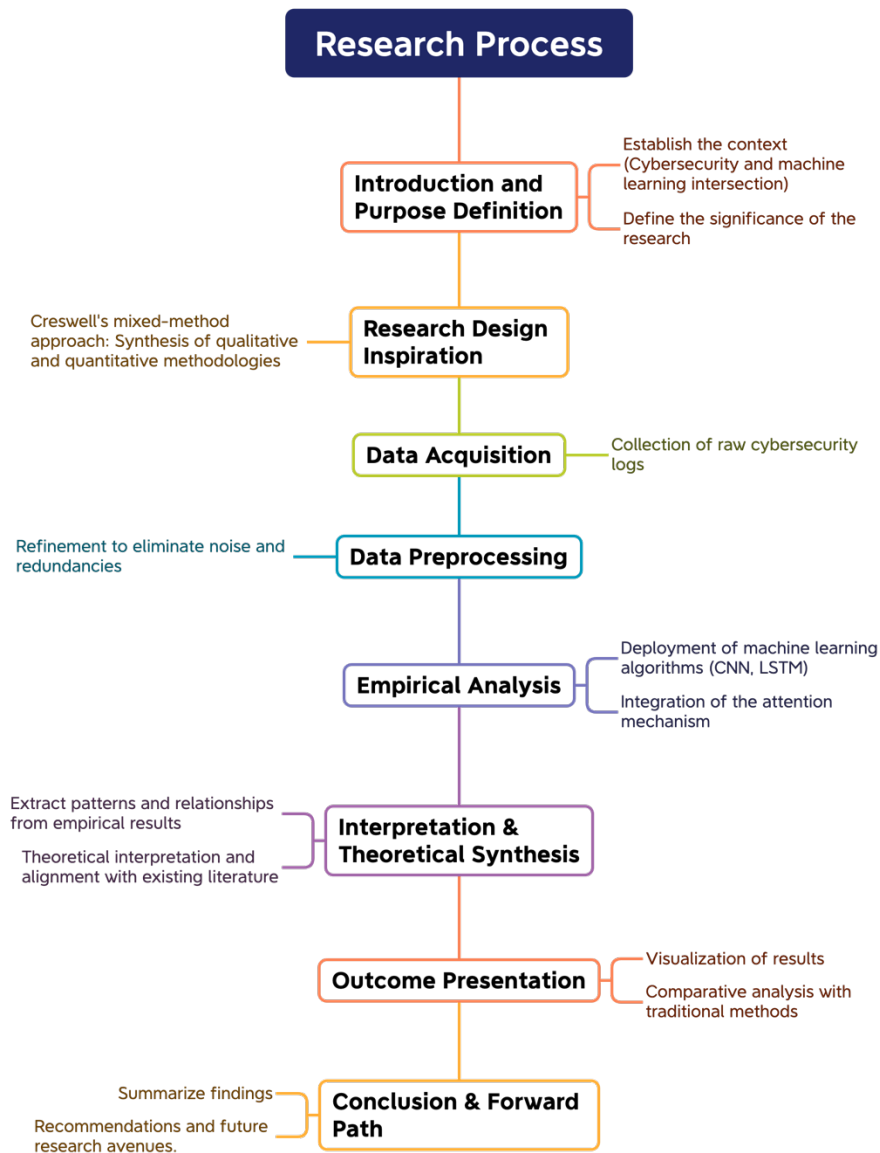


Figure 1: Research Process

2.2.2 Research Philosophy

Diving deeper into the foundational beliefs that underpin this research, we find ourselves nestled within the paradigm of post positivism objectivity. This philosophy, though rooted in the traditional positivist notion of an objective reality, acknowledges that our understanding of this reality is always imperfect and probabilistic. When applied to the intricate landscape of cybersecurity, post

positivism emerges as an advocate for critical realism. It suggests that while there is an objective cybersecurity reality, comprising threats, vulnerabilities, and incidents, our understanding of this reality is often mediated by our tools, methodologies, and interpretative lenses.

The emphasis on empirical data within this philosophical framework becomes self-evident. In the vast and intricate domain of cybersecurity, where threats metamorphose rapidly and novel challenges emerge with alarming regularity, reliance on theoretical postulations alone is akin to navigating treacherous waters with an outdated map. Empirical data, characterized by its raw, unmediated nature, serves as the North Star. It offers researchers real-time insights into the evolving nature of threats, ensuring that our predictive models remain both relevant and effective.

Furthermore, the importance of empirical data extends beyond mere data-driven insights. In the world of predictive analytics for cybersecurity, the veracity and efficacy of models hinge heavily on the quality and timeliness of data. Here, empirical data is not just a passive informant; it plays an active role in sculpting, refining, and validating models. It ensures that these models, when deployed in real-world scenarios, perform with a degree of accuracy and reliability that aligns with the dynamism of the cybersecurity landscape. In essence, the choice of post positivism objectivity as the guiding philosophy is a testament to the research's commitment to aligning with the objective realities of cybersecurity, all while acknowledging the inherent limitations and constantly striving for refinement through empirical evidence.

2.2.3 Research Methodologies

In an academic sphere characterized by a rich tapestry of methodologies, this research consciously gravitates towards a mixed-methods approach. This inclination, rather than being a mere academic fancy, is a deliberate attempt to harness the multifaceted nature of cybersecurity challenges. Within the quantitative ambit, the primary focus is on model accuracy and evaluations. The objective metrics derived from these analyses illuminate the model's performance,

emphasizing aspects like precision, recall, and the overarching fidelity to real-world scenarios. Such a quantitative orientation ensures that the research is anchored in empirical reality, with models that are both robust and reliable.

In parallel, a qualitative lens is employed, not as an adjunct but as a critical cog in the research machinery. Cybersecurity, with its intricate interplay of patterns, anomalies, and ever-evolving threat vectors, demands a nuanced understanding that transcends mere numbers. The qualitative approach, in this context, becomes a beacon, shedding light on subtle patterns, emergent anomalies, and the underlying narratives that might escape a strictly quantitative purview. Not only does this approach facilitate a deeper comprehension of the threat landscape, but it also fosters a more holistic view, weaving together the disparate threads of cybersecurity into a coherent tapestry.

Central to this methodology are the UNSW-NB15 datasets, which hold a distinctive place in cybersecurity research. This dataset isn't just a repository of information; it is, in many ways, a microcosm of the broader cybersecurity milieu. Its significance stems from its comprehensive nature, encompassing a myriad of attacks, behaviors, and patterns representative of real-world scenarios. Moreover, its scope in terms of volume and variety makes it an indispensable asset for any empirical analysis in this domain.

Delving deeper into these datasets reveals qualitative insights that have the potential to transform and inform the quantitative model development process. For instance, understanding the narrative behind specific anomalies or recognizing emergent patterns can guide the feature extraction process, ensuring that models are trained on pertinent data points. Furthermore, qualitative interpretations of these datasets can elucidate potential biases, inform model refinement, and even inspire novel methodologies. In essence, the interplay between the qualitative insights from these datasets and the quantitative rigor ensures that the research remains both grounded in reality and attuned to the subtleties of the cybersecurity domain. This synergy, where numbers meet narratives, is the fulcrum on which the research methodologies pivot, promising insights that are both profound and actionable.

3.2.4 Approach

Within the vast expanse of research methodologies, this study adopts a deliberate stance by gravitating towards a mono method approach. Instead of dabbling in diverse methods and potentially diluting the analytical focus, the research anchors itself in a pure analytical model-based approach specifically designed for threat detection. Such an approach, when viewed from a peripheral perspective, might seem restrictive. However, upon deeper introspection, its merits become increasingly apparent.

The realm of cybersecurity is characterized by an incessant metamorphosis. Threat vectors, once thought to be vanquished, often resurface with renewed vigor and unprecedented sophistication. In this dynamic landscape, traditional heuristic-based detection methods often find themselves in a perpetual game of catch-up, always one step behind the adversary. Herein lies the justification for a mono method, analytical model-based approach: its innate ability to adapt, learn, and preemptively address threats.

The ever-evolving nature of these threats doesn't just warrant but actively demands empirical methods like machine learning. Machine learning models, characterized by their self-learning capabilities, are uniquely positioned to recognize and adapt to new threats, all the while refining their detection algorithms. This is not a mere enhancement over traditional methods but a paradigm shift, ushering in a new era of proactive, rather than reactive, threat detection.

Furthermore, the sheer volume and velocity of data in the contemporary digital ecosystem necessitate an approach that can process, analyze, and act in real-time. The mono method approach, grounded in analytical modeling, is tailor-made for this challenge. Not only does it ensure that the threat detection mechanism is robust, but it also guarantees that it remains agile, scalable, and future-ready.

In essence, the choice of a mono method approach is not a matter of convenience but a conscious decision borne out of a comprehensive understanding of the

cybersecurity domain. It's a testament to the research's commitment to staying ahead of the curve, leveraging cutting-edge methodologies to ensure that the digital realm remains secure, even in the face of ever-evolving threats.

3.2.5 Strategies

Embarking on a journey into the intricate realms of cybersecurity necessitates a strategy that is both robust and adaptable. Central to this strategy is a comprehensive understanding of the datasets under scrutiny. Recognizing the myriad facets of these datasets isn't merely an academic exercise; it's pivotal to ensuring the research remains grounded, relevant, and poised to offer actionable insights. These datasets, encapsulating vast volumes of cyber activities, serve as the foundational bedrock upon which the analytical edifice of this research stands.

Yet, a mere understanding of the datasets is only half the battle. The true mettle of this research is manifested in its dedication to model refinement, a process that seamlessly marries qualitative insights with quantitative rigor. By delving deep into network patterns, the study aims to unearth the underlying narratives, the subtle nuances, and the emergent trends that often elude a cursory glance. These qualitative insights, when synergized with quantitative data, particularly the stark empirical evidence of anomalies, yield a model that is both reflective of the ground realities and predictive of emergent threats.

Furthermore, the strategy doesn't stop at merely crafting an innovative detection model. In a bid to ascertain its efficacy, a comparative analysis has been instituted, pitting the proposed method against the bulwark of conventional intrusion detection systems. Such a comparative endeavor serves a dual purpose. First, it places the research in a broader context, allowing for an evaluation against established benchmarks. This juxtaposition, in turn, highlights the merits, and potential areas of improvement, of the proposed model. Second, this comparison acts as a bridge, linking the avant-garde methodologies of the research with the tried-and-tested paradigms of the past. Not only does this foster a dialogue between the old and the new, but it also ensures the research remains both forward-looking and rooted in historical precedent.

3.2.6 Choices

Navigating the labyrinthine corridors of cybersecurity research often demands a clear articulation of choices, for these choices determine the trajectory and the eventual outcomes of the inquiry. Central to this research's strategic choices is an unequivocal emphasis on deep learning models, an avenue that promises both depth and precision. But the question arises: Why deep learning? And more specifically, why the trio of CNN (Convolutional Neural Networks), LSTM (Long Short-Term Memory), and the attention mechanism?

The rationale is multifaceted. First, CNNs, by their very design, excel in feature extraction. In the ever-complex world of cybersecurity, where threats manifest in various guises, the ability to extract salient features from vast troves of data becomes paramount. The introduction of LSTM then takes the baton from CNN, focusing on sequence prediction. Cybersecurity isn't just about identifying threats but predicting their sequence, understanding their patterns, and pre-empting their next move. LSTMs, with their memory cells, are adept at handling such sequences, especially in time-series data, making them invaluable for the task at hand.

Deep learning has its own challenges. The sheer volume of data can often introduce noise, obscuring the signals. This is where the attention mechanism enters the fray, offering a salient solution. By dynamically allocating weights, the attention mechanism ensures that the model focuses on what truly matters, mitigating extraneous distractions. It's akin to a vigilant sentinel, always on the lookout, ensuring that the focus remains undeterred.

However, the choices extend beyond the selection of models. At its core, this research endeavors to strike a harmonious balance between qualitative insights and quantitative model evaluation. While deep learning models offer empirical rigor, they're further enriched by qualitative interpretations. Insights drawn from network patterns, user behaviors, and historical contexts offer a richer tapestry, one that ensures the models remain grounded, contextual, and, most importantly, relevant.

3.2.7 Time Horizons

The time dimension, while often overlooked, can significantly influence the outcomes and interpretations of any research. Particularly in domains as dynamic as cybersecurity, where threats evolve, adapt, and mutate with alarming alacrity, the choice of time horizons becomes all the more critical. In the case of this study, an informed decision was made to adopt a cross-sectional approach, shedding light on specific temporal frames. The rationale for this choice demands deeper introspection.

Deep learning, for all its transformative potential, is inherently iterative. Models learn, adjust, and refine their understanding over numerous cycles, harnessing the power of data to gradually enhance their predictive prowess. Given this iterative nature, a longitudinal approach, while comprehensive, might dilute the specific nuances associated with each iteration. Therefore, the choice of a cross-sectional approach emerges as a calculated strategy, designed to capture the richness of these iterative cycles.

By focusing on distinct time frames during which data was collected and analyzed, this approach offers a snapshot, a temporal freeze-frame if you will, of the model's performance, challenges, and evolution. Not only does this provide a clearer perspective on how the model adapts over time, but it also ensures that specific patterns, unique to particular time frames, are not lost in the broader sweep of a longitudinal analysis.

Moreover, given the dynamism of cybersecurity threats, a cross-sectional perspective can illuminate transient threats, fleeting vulnerabilities, and ephemeral patterns that might be obscured in a more extended timeline. This ensures that the research remains agile, responsive, and attuned to the subtle ebbs and flows of the cybersecurity landscape.

3.2.8 Techniques And Procedures

In the vast seas of cybersecurity logs, data, in its raw form, is often riddled with inconsistencies, redundancies, and noise. Such raw data can mislead and skew the findings if consumed unfiltered. Therefore, the research employs a suite of data preprocessing techniques, ensuring that the dataset of UNSW-NB15 is not only consistent but also tailored to enhance the efficacy of the deep learning models. These techniques range from normalization and feature selection to handling missing values and outlier detection. This refined data serves as the bedrock, feeding the sophisticated architectures of CNNs and LSTMs.

Following the preprocessing stage, the research delves into the nuanced realms of model training, validation, and testing. This triad, while standard in nomenclature, is distinctly tailored for this study. Recognizing the intricacies of cybersecurity threats, the model training is iterative, learning from both the patterns and anomalies within the datasets. Additionally, the validation steps are more than mere checkpoints. They are instrumental feedback loops, guiding the model, refining its parameters, and ensuring its readiness for the testing phase. The testing, in turn, is not a culmination but a rigorous assessment, a litmus test for the model's real-world applicability.

But what truly sets this research apart is its nuanced approach to performance evaluation. In an era dominated by quantitative metrics, this study acknowledges the need for balance. While conventional metrics like accuracy, precision, and recall are undoubtedly pivotal, they alone cannot capture the holistic performance of a cybersecurity predictive model. Therefore, the research harmoniously integrates both quantitative metrics and qualitative insights, offering a panoramic view of the model's efficacy. Through this dual-lens evaluation, the study ensures that not only are the numbers sound, but the narrative, the story that these numbers tell, is cogent, comprehensive, and compelling.

3.3 CONCLUSION

Throughout this chapter, we've endeavored to provide a holistic insight into the many facets of our research journey. We commenced with an elucidation of the

research's intent, emphasizing the integration of machine learning with cybersecurity. A visual representation painted a clear picture of the entire process. Philosophical underpinnings laid the groundwork, emphasizing post-positivism objectivity tailored for cybersecurity. Our methodologies demonstrated a preference for mixed methods, guiding readers through our selection and utilization of dataset like UNSW-NB15.

Our approach, strategic considerations, and choices threw light on the intricate mesh of decisions that drove this research, with a particular focus on deep learning models like CNNs, LSTMs, and the innovative attention mechanism. Recognizing the temporal nuances of our endeavor, we highlighted our adherence to a cross-sectional timeline, driven by the iterative cadence of model training. Lastly, the chapter delineated our data-handling procedures, ranging from preprocessing to validation and evaluation, emphasizing a harmonized use of both qualitative and quantitative metrics. In essence, this chapter serves as a mirror, reflecting our commitment, choices, and the scientific rigor underscoring our expedition into predictive analytics for cybersecurity.

CHAPTER 4 - DESIGN & IMPLEMENTATION

4.1 INTRODUCTION

This Chapter focuses on the design and implementation processes crucial for predictive analytics in threat intelligence within cybersecurity log management. This chapter introduces the problem we aimed to solve and then systematically unveils the architecture of the model. We will discuss the amalgamation of Convolutional Neural Networks (CNN) for spatial feature analysis and Long Short-Term Memory (LSTM) for temporal feature integration. A significant emphasis will be placed on the innovative integration of the self-attention mechanism. This approach aims to enhance the accuracy of intrusion detection by dynamically adjusting connection weights and efficiently handling variable-length input sequences. Furthermore, the chapter elucidates the practical steps for implementation, including data preprocessing from the UNSW-NB15 dataset, constructing the neural network model, optimization strategies, and the environment suitable for deploying the model. Our primary focus throughout this chapter remains on how these methodologies can elevate the analytical and predictive prowess in network security log management.

4.2 PROBLEM FORMULATION

In the burgeoning realm of cybersecurity, the proliferation of sophisticated cyber-attacks necessitates the development of advanced intrusion detection systems. The scale and intricacy of network traffic in large-scale networks demand predictive analytics tools that can discern both spatial and temporal features, ultimately enhancing threat intelligence. In order to achieve the main aim of this paper, we proposed the pivotal question driving this research is:

How can a deep neural network architecture effectively harness the spatial and temporal characteristics of network traffic to augment intrusion detection in large-scale networks?

In this project, we divide this main problem into several sub-problems:

1. **Spatial Feature Extraction:** In the vast swathes of network traffic data, the extraction of pertinent spatial features is paramount. Can a CNN effectively learn and extract these high-level features without getting overwhelmed by the sheer volume of data?
2. **Temporal Feature Understanding:** Beyond spatial features, the sequencing of network activity is a critical dimension in predicting potential intrusions. How can LSTMs be integrated after CNN to not just comprehend but predict temporal sequences in the network traffic?
3. **Integration of Self-Attention Mechanism:** With the copiousness of data in network traffic, discerning the crucial segments from the inconsequential ones is essential. How can the self-attention mechanism accentuate the model's focus, dynamically weighing different data points to bolster the detection accuracy?

Then, our proposed approach is utilizing an avant-garde deep neural network architecture, grounded in a series of combination blocks that meld the strengths of CNN, LSTM, and Self-Attention:

1. **CNN for Spatial Learning:** By initiating each combination block with a CNN, the model aims to derive high-level spatial features from traffic data. The architecture employs various operations like convolution, pooling, and batch normalization to optimize the learning process.
2. **LSTM for Temporal Learning:** Each CNN's output is channeled into an LSTM within the same combination block. This ensures that spatially extracted features are temporally sequenced, offering a holistic view of network traffic patterns.
3. **Layered Self-Attention Integration:** In a groundbreaking move, this model leverages the self-attention mechanism, not just as an ancillary component but as a fundamental layer within the architecture. This allows the model to dynamically assign weights to variable-length input sequences, isolating and emphasizing the most pertinent data segments. The intricate design of the hidden layers, progressing from 64 filters/units in the initial level to a whopping 256 in the final tier, underscores the model's commitment to learning granularity. As the architecture ascends

in complexity, it promises a meticulous and nuanced understanding of intrusion patterns.

4.3 DESIGN

4.3.1 The Overall Architecture

As shown in the Figure 2, the architecture embraces a sophisticated amalgamation of convolutional neural networks (CNN), long short-term memory networks (LSTM), and Self-Attention mechanisms. The holistic approach not only aims to harness the strengths of each individual component but also synergizes them to work in unison, fortifying the entire structure.

The rationale behind our architectural choice is multifaceted. And a very important work we learned from our predecessors is LuNet (Wu & Guo, 2019), which supports this idea. Firstly, the CNN is particularly adept at learning spatial features inherent in traffic data. Given the vast and varied nature of network traffic, leveraging the power of CNN ensures that spatial relationships within this data are captured and utilized effectively. On the other hand, the LSTM brings its prowess in learning temporal patterns to the table. As network activities often exhibit temporal dependencies, the LSTM component allows the model to capture and remember these long-term dependencies, making the model highly perceptive to nuanced patterns indicative of intrusions. In addition to the strengths of CNN and LSTM, the introduction of the Self-Attention mechanism serves as a linchpin in enhancing the model's discriminative power. It dynamically assigns weights to various connections, ensuring that the model remains attuned to the most crucial pieces of information, while simultaneously diminishing the influence of extraneous data.

Further elucidation on the need for a composite model emerges from the intricate nature of network data. While CNNs and LSTMs individually bring significant advantages, their combined potential remains unparalleled. In our architecture, the CNN operates upstream of each RNN level, ensuring that high-level feature

extraction precedes temporal pattern recognition. This harmonized learning at every level ensures minimal interference between the two, optimizing the granularity of learning based on the intricacies of both spatial and temporal patterns. Moreover, with the overlay of the Self-Attention mechanism, the architecture gains the ability to fine-tune its focus, zeroing in on paramount segments of data.

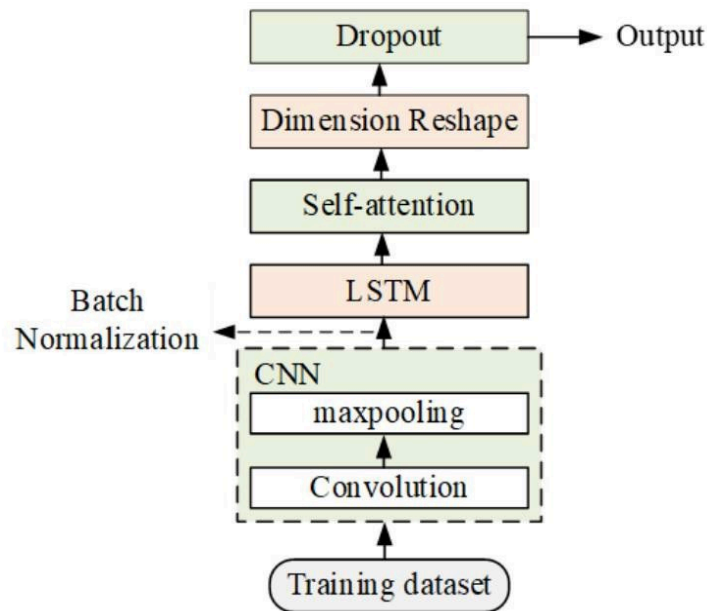


Figure 2: The Architecture of Model

4.3.2 Spatial Feature Analysis with CNN

Understanding network traffic often involves deciphering intricate patterns within a vast array of data. Among the various types of patterns, spatial features stand out as vital components. Essentially, spatial features in network traffic highlight the relationships and patterns of data points when they are closely knit together. It's analogous to identifying patterns on a canvas by looking at the arrangement and relationship of colors and shapes in close proximity. In the context of intrusion detection, these spatial features can be indicative of abnormal traffic patterns, suspicious data transfers, or even the initial stages of cyber-attacks.

Convolutional Neural Networks (CNNs) have historically been an instrumental tool in image processing, primarily due to their proficiency in extracting spatial

hierarchies. In the domain of network traffic analysis, CNNs operate under a similar principle. The main operation behind CNNs is convolution. Mathematically, convolution involves taking the dot product of a small matrix, known as the filter or kernel (K), and a portion of the input data (I), producing a feature map (F).

$$F(i, j) = \sum_m \sum_n I(i + m, j + n) \cdot K(m, n)$$

This formula showcases the convolution operation where each element of the feature map is computed by iterating over the input data and the kernel. Through this process, CNNs are able to detect local patterns in the data.

A deeper design specifics:

Convolution layers

At the heart of the CNN, these layers utilize filters that move across the input data (such as a time slice of network traffic) to produce feature maps. By using multiple filters, the network can identify a plethora of spatial patterns, each represented by a distinct feature map.

Pooling layers

Following the convolution layers, pooling layers are employed to reduce the spatial dimensions of the feature maps, all while retaining the most critical information. The most common type of pooling is max pooling. Given a region in a feature map, max pooling simply selects the highest value from that region. Mathematically, for a 2x2 region:

$$\text{MaxPool}(x) = \max(x_{i,j})$$

where $x_{i,j}$ represents the elements in the region.

Batch normalization

In the process of training deep neural networks, internal covariate shift can occur, slowing down the training process and requiring careful initialization. Batch normalization mitigates this by normalizing the activations of each layer. Given activations x , the operation is:

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}}$$

where μ is the mean of x , σ^2 is the variance, and ϵ is a small constant to avoid division by zero.

In conclusion, CNNs, with their intricate design of convolution, pooling, and normalization layers, provide a powerful means to dissect and interpret spatial patterns in network traffic. This ability to discern patterns at multiple levels of granularity equips our system with a robust mechanism to detect anomalies and intrusions, laying the foundation for the subsequent temporal analysis by the LSTMs.

4.3.3 Temporal Feature Analysis with LSTM

Intrusion detection in network traffic isn't solely about understanding the spatial relationships between individual data points; it's equally about appreciating the sequences in which these data points appear. While spatial features enable us to capture patterns in closely clustered data, temporal sequences help identify patterns across time. Such sequences are crucial in detecting prolonged or sophisticated attacks that manifest over time. For instance, understanding the sequence of network requests can reveal patterns associated with a potential breach, even if individual requests might appear benign.

Long Short-Term Memory networks (LSTM) are a subset of recurrent neural networks (RNNs) specifically designed to handle long-term dependencies and sequences. Classic RNNs struggle to retain information over extended sequences due to the vanishing gradient problem. LSTMs address this limitation through their unique architecture comprising of input, forget, and output gates.

The core concept behind LSTMs is to allow information to be added or removed from a cell state. The gates within an LSTM unit decide the extent of information to

be passed on. Mathematically, these operations are represented by sigmoid and tanh functions:

Forget gate

Determines the amount of information to discard from the cell state.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Input gate

Updates the cell state with new information.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

Cell state update

Combines the operations from the forget and input gates.

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t$$

Output gate

Determines the final output of the LSTM unit.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$

$$h_t = o_t \times \tanh(C_t)$$

Where:

- f_t, i_t, o_t are the activations of the forget, input, and output gates, respectively.
- σ denotes the sigmoid function, ensuring gate activations stay between 0 and 1 .
- W and b are the weights and biases for each gate.
- h_{t-1} is the previous hidden state, and x_t is the current input.
- C_t is the updated cell state, and h_t is the current hidden state.

Here is the detailed design layout.

LSTM layers

The architecture starts with a foundational LSTM layer designed to capture basic temporal patterns. As one progresses deeper, the layers are meant to recognize and retain increasingly complex temporal relationships.

Units and their complexity

In the initial layers, the LSTM units are fewer (e.g., 64 units). As we ascend the layers, the complexity increases, which is reflected by an increasing number of LSTM units, peaking at 256 in the final layers. The rationale behind this progression is twofold: first, initial layers can swiftly capture rudimentary patterns with fewer parameters, and second, deeper layers, with their increased capacity, are geared to encapsulate intricate temporal dependencies that become evident after preliminary patterns have been discerned.

In essence, the LSTM's unique capability to remember and process long-term sequences equips our model to understand temporal nuances within the network traffic. By discerning patterns over time, combined with the spatial insights from CNN, the model stands as a potent tool in intrusion detection, ensuring both immediate and prolonged threats are aptly recognized.

4.3.4 Integration of Self-Attention Mechanism

In the vast realm of deep learning, attention mechanisms have paved a transformative path. These mechanisms stem from the idea that, while processing sequences, not all parts are of equal relevance. Some portions need more "attention" than others, and recognizing these pertinent fragments is vital for improved model accuracy.

At its core, the attention mechanism seeks to assign a weight to each input feature, based on its significance. Instead of processing every part of the input sequence with equal importance, the mechanism lets the model dynamically "focus" on the most relevant parts, akin to human selective attention.

The Self-Attention mechanism, an evolution of this idea, calculates the relevance or attention scores of different parts of a sequence relative to every other part.

This self-referential mechanism allows a richer understanding of sequence data, as the relevance of each data point is judged in the context of the entire sequence.

To elucidate, consider an input sequence X . The Self-Attention mechanism computes three entities from X : queries (Q), keys (K), and values (V):

$$\begin{aligned}Q &= W_Q \cdot X \\K &= W_K \cdot X \\V &= W_V \cdot X\end{aligned}$$

Where W_Q , W_K , and W_V are trainable weight matrices.

The attention weights are calculated using a *softmax* function over the dot product of the queries and keys, scaled down by the square root of the depth d_k :

$$Attention(Q, K, V) = \text{softmax} \left(\frac{K^T Q}{\sqrt{d_k}} \right) V$$

In our architecture, the Self-Attention layer follows the LSTM layer. Once the LSTM processes the input data and captures its temporal dynamics, the Self-Attention mechanism refines these dynamics. It allows the model to discern not just what information is present, but also the relative importance of this information in the context of the entire sequence. This sequencing ensures that the temporal patterns, once captured by LSTM, are then weighted for their significance, providing a richer, more nuanced feature set for subsequent layers or outputs.

Integrating the Self-Attention mechanism after LSTM layers offers several compelling benefits. Firstly, LSTM, being adept at retaining long-term dependencies, provides the model with the temporal patterns present in the data. Following this with Self-Attention ensures that these patterns are not just captured, but are also weighted based on their importance. This is particularly crucial for intrusion detection, where certain patterns, though temporally distant, might be critically interlinked.

Furthermore, as the Self-Attention mechanism allows the model to consider each part of the sequence in relation to every other part, it provides a rich contextual

understanding of network traffic. This is invaluable in detecting sophisticated attacks where context, rather than individual actions, might indicate a breach.

In conclusion, the placement of the Self-Attention mechanism post LSTM is a deliberate design choice aimed at maximizing the understanding of network traffic data. By first capturing temporal dynamics and then weighing them based on contextual significance, the architecture ensures a robust, detailed, and context-aware analysis, critical for effective intrusion detection.

4.3.5 Hidden Layer Design

Within neural network architectures, hidden layers are central to a model's capacity to extract features and establish intricate patterns. They act as the intermediate zones where raw inputs are transformed through various operations, resulting in a representation that facilitates the task at hand. For our architecture, which seeks to detect intrusions in network traffic, hidden layers are designed to discern both the spatial and temporal intricacies.

Hidden layers, placed between the input and output layers, provide a space for the network to learn and encode complex patterns from the data. Their neurons apply transformations to the inputs, and the depth and complexity of these layers determine the model's capacity to learn intricate patterns. In the context of our model, the hidden layers are responsible for dissecting the input data, extracting critical features, and refining them for eventual intrusion detection.

Our architecture employs a progressive increase in complexity, starting with 64 filters/units and culminating at 256 in the final layers. This escalation is strategically significant.

Mathematically, given an input volume of size $W_1 \times H_1 \times D_1$, applying a convolutional layer with $F \times F \times D_1$, the output volume size (assuming stride 1 and no padding) is $W_2 \times H_2 \times K$ where:

$$W_2 = W_1 - F + 1$$
$$H_2 = H_1 - F + 1$$

The initial layers with 64 filters/units focus on capturing low-level features. As the input progresses through the layers, these features are combined and refined, demanding more filters to capture the increasing complexity. By the time the data reaches the layers with 256 filters/units, the model is processing high-level, abstract features that are crucial for accurate intrusion detection.

Moreover, speaking of the granularity of learning, it is determined by the architecture of the hidden layers. Granularity here refers to the depth and detail of the information captured from the input data. As the number of filters/units increases progressively, each layer becomes capable of dissecting the data with more precision.

For instance, initial layers with 64 filters might identify basic patterns like packet intervals or common headers in network traffic. As we move deeper into the layers with increased filters, the network begins recognizing combinations of these patterns, such as specific sequences of packet transfers that might signify an intrusion attempt.

The granularity also gets impacted by other design components like convolution operation, pooling, and batch normalization. Pooling layers, for example, reduce the spatial dimensions while retaining crucial information, which directly influences the granularity by making the subsequent layers focus on broader (but important) patterns.

Basically, our hidden layer design embodies the principle of "from general to specific." The increasing complexity, represented by the growing number of filters/units, ensures that as data moves through the network, its representation becomes more refined, detailed, and attuned to the intricacies essential for effective intrusion detection. This design choice, rooted in the understanding of how network data behaves and what's needed to detect anomalies, ensures a balance between model efficiency and its detection capabilities.

4.3.5 Combination Blocks

At the heart of our neural network architecture lie the combination blocks—well-orchestrated symphonies of CNN, LSTM, and Self-Attention components, further enhanced with dimension reshaping and dropout functionalities. Each element, optimized for a specific facet of learning, together promises an all-rounded intrusion detection system.

The journey begins with the CNN component. Entrusted with extracting spatial features, the CNN navigates network traffic, analogous in complexity to image data, for spatial correlations. Filters in the CNN slide over input data, diligently identifying patterns, whether they manifest as regular packet exchanges or recognizable data structures. The convolutional layers of this component thus offer an initial mapping of spatial nuances intrinsic to the data.

Once the spatial context is gleaned, the baton passes to the LSTM component. Here, the temporal architecture of network traffic is decrypted. Given the inherently sequential nature of network activities, an LSTM—with its specialized memory units—traces patterns across time, weaving seemingly isolated events into a larger narrative of network behavior. The gaps between data packets, their arrival sequences, and other time-bound patterns are meticulously documented, facilitating the spotting of anomalies.

Next, we delve into the Self-Attention mechanism. Amidst the vastness of network data, certain sequences and features stand out in their relevance. This is where Self-Attention comes into play, dynamically allocating weightage to different portions of input data based on their significance. By focusing its lens on these pivotal data fragments, the mechanism elevates the model's precision, filtering distractions and amplifying potential threats.

Post the Self-Attention phase, the data is subject to dimension reshaping. Given that layers and mechanisms may have varied output dimensions, it's imperative to maintain a consistent data shape as it flows from one component to the next.

Reshaping ensures that the output from the Self-Attention mechanism aligns with the expected input dimensions of subsequent layers or blocks, ensuring seamless integration and data flow.

Following reshaping, a dropout stage is introduced. In the quest for an optimal neural network, the risk of overfitting—where a model becomes too attuned to training data and loses generality—is ever-present. Dropout serves as a regularization technique, randomly deactivating a subset of neurons during training. This not only promotes a more generalized model but also enhances its robustness, making it resilient to potential noise in the data.

Data's journey through these combination blocks is both linear and cyclical. Starting from the CNN for spatial extraction, progressing to the LSTM for time-bound insights, refined by Self-Attention, reshaped for dimensional consistency, and finally subjected to dropout for generalization—the data either culminates as an output or seamlessly integrates into the next combination block.

The learning within these blocks echoes the principle of iterative refinement. Each traversal sharpens the model's acuity, allowing it to evolve from detecting blatant anomalies to identifying nuanced threats. This tiered learning approach ensures that the model remains comprehensive in its threat detection, adeptly handling both common intrusions and emerging, sophisticated challenges.

4.4 IMPLEMENTATION

4.4.1 Data Preprocessing

Embarking on our journey into this dataset UNSW-NB15 (Moustafa & Slay, 2015), we acquaint ourselves with its basic information. Housed within are a staggering 82,332 entries meticulously categorized across 45 columns. Delving deeper, we identify a tapestry of data types: 11 columns dancing to the tune of floats, a dominant presence of 30 integer columns, and a sprinkle of 4 columns adorned with objects, often strings or categorical in nature.

Transitioning from the broad overview to a microscopic perspective, an examination of the first few rows unravels a plethora of insights. The dataset elegantly weaves together a blend of numerical and categorical features. Among the myriad of columns, the likes of proto, service, and state clearly flaunt their categorical badges. In addition, nestled within this vast landscape, we discern the attack_cat column, a sentinel alluding to the type of cyber onslaught, and its compatriot, the label column, which whispers hints of its role in binary classification.

Our thirst for understanding deepens, leading us to the sanctum of summary statistics. Through this lens, the numerical columns regale tales of mean, standard deviation, extreme values, and more. Not to be overshadowed, the categorical realms of proto, service, state, and attack_cat showcase their own narratives, detailing their most frequent visitors and the frequency of their sojourns. Among these tales, a few narratives stand out: the label column's duality, oscillating between 0 and 1; the rich diversity of the attack_cat column, boasting 10 unique tales with "Normal" being the crowd favorite; and the sweeping ranges observed in many a numerical column, hinting at a tapestry of varied distributions.

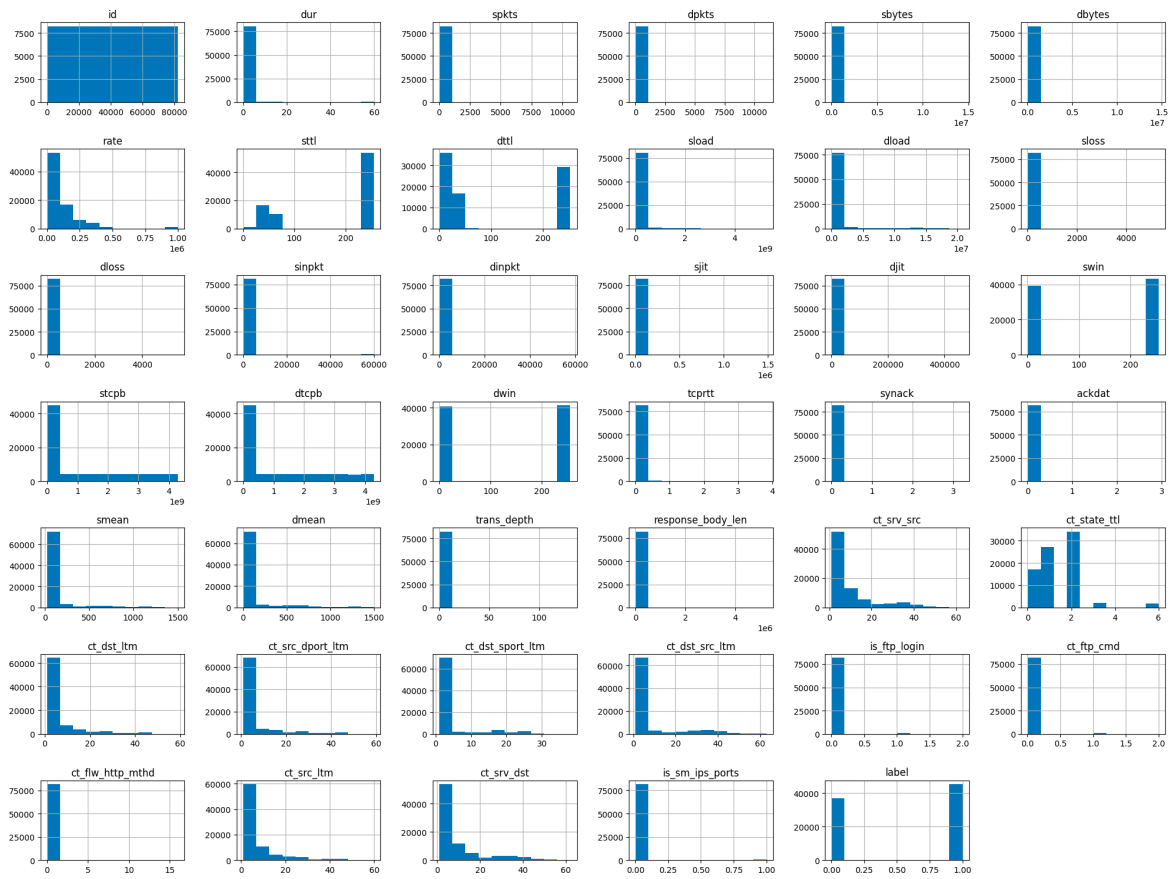


Figure 3: Histograms for numerical columns

But numbers, while enlightening, often yearn for a visual companion. This beckons the entrance of histograms, as shown in the Figure 3. These graphical narrators reveal skewed stories in protagonists like dbytes, sbytes, rate, and sttl, all leaning heavily towards their narrative's beginning. Simultaneously, certain columns such as ct_ftp_cmd and is_ftp_login march to their own beat, displaying distinct, discrete tales. Moreover, the label column, in its visual narrative, paints a harmonious balance between its dual protagonists, 0 and 1.

Yet, as any seasoned data aficionado knows, understanding individual narratives is but half the battle. The magic often lies in the interplay, the correlations, the secret dances between these columns. Enter the correlation heatmap, as shown in the Figure 4. Acting as our guide is a vibrant palette: warm shades leaning towards red heralding tales of positive camaraderie, cooler hues inching towards blue speaking of distant relationships, and the neutral, balanced whites whispering stories of independence. Among these colorful tales, the duet of sbytes and spkts

emerges as a heartwarming tale of strong, positive companionship. In contrast, a few stories touch upon themes of separation and distant relationships, though they are few and far between.

Before embarking on our transformative journey, a quick pitstop is made at the sanctuary of missing values. Our spirits are uplifted with the revelation of an impeccable dataset, devoid of any voids, ensuring our preprocessing path remains unobstructed.

As we gear up for data preprocessing, our plan unfolds in layers. Categorical columns, particularly proto, service, and state, undergo a metamorphosis, emerging as one-hot encoded vectors. Furthermore, outliers, those pesky narrative disruptors, are gracefully handled with a double-sided log transformation. Rounding off our preprocessing saga, a sigmoid transformation ensures each data value finds its home between 0 and 1.

With the transformation complete, our eyes turn towards the processed data, keenly inspecting the fruits of our labor, and setting the stage for the analytical ballet that lies ahead.

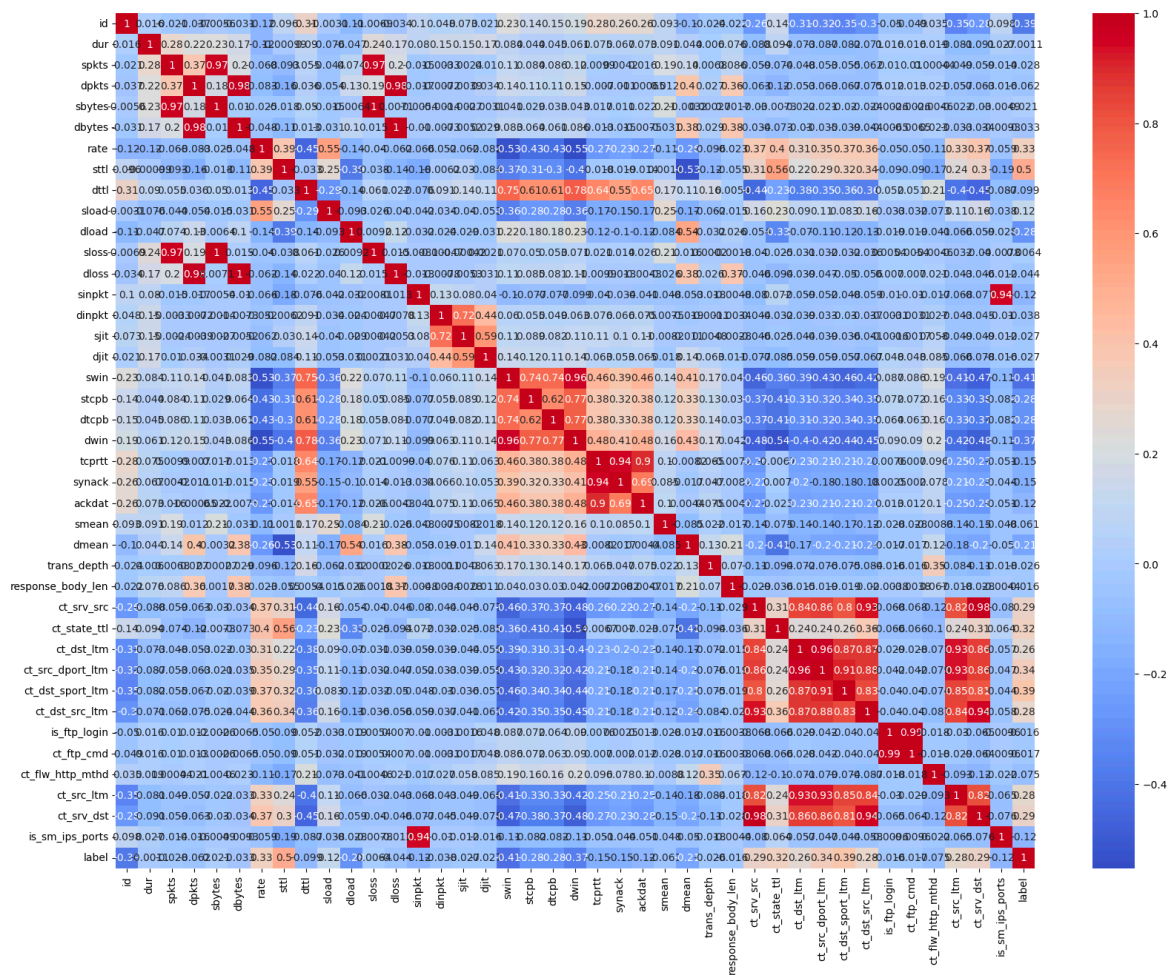


Figure 4: Correlation Heatmap for Numerical Columns

4.4.2 Building the Neural Network Model

Harnessing the strengths of the right software and libraries is significant to building effective neural network architectures. The provided code leans heavily on the robust PyTorch framework, supplemented with the PyTorch Lightning wrapper. Not only does PyTorch offer a dynamic computation graph, which is ideal for flexible neural network designs, but when combined with PyTorch Lightning, it streamlines the training process, making it more efficient and user-friendly.

Within this structure, the Combination Block Implementation (Figure 5) plays a crucial role. At the heart of the architecture lies the Convolutional Neural Network (CNN) layers designed meticulously for spatial feature extraction. These layers are adept at capturing the intricate spatial patterns in the data. In addition to the CNNs,

LSTM layers are integrated to capture the temporal patterns in sequences, ensuring that the model understands the progression and dependencies in data over time.

```
class LuNetBlock(LightningModule):
    def __init__(self, hparams):
        super().__init__()
        self.conv = nn.Sequential(
            nn.Conv1d(
                in_channels=1,
                out_channels=hparams.conv_out,
                kernel_size=hparams.kernel
            ),
            nn.ReLU()
        )
        self.max_pool = nn.Sequential(
            nn.MaxPool1d(kernel_size=2),
            nn.ReLU()
        )
        # calc the output size after max_pool
        dummy_x = torch.randn(1, 1, hparams.input_dim, requires_grad=False)
        dummy_x = self.conv(dummy_x)
        dummy_x = self.max_pool(dummy_x)
        max_pool_out = dummy_x.shape[1]
        lstm_in = dummy_x.shape[2]

        self.batch_norm = nn.BatchNorm1d(max_pool_out)

        self.lstm = nn.LSTM(lstm_in, hparams.conv_out)
        self.selfAtt = Self_Attention(input_dim=hparams.conv_out, dim_k=hparams.conv_out, dim_v=hparams.conv_out)
        self.dropout = nn.Dropout(p=0.5)

    def forward(self, x):
        x = self.conv(x)
        x = self.max_pool(x)
        x = self.batch_norm(x)

        # lstm and relu+selfAttention
        x, hidden = self.lstm(x)
        x = self.selfAtt(x)
        .. = r +--+--+..
```

Figure 5: The Combination Block Implementation Code

But the model doesn't stop there. A sophisticated Self-Attention mechanism (Figure 6) is introduced, adding another layer of intricacy to the model. This mechanism allows the model to weigh the importance of different parts of the input data, enhancing its capacity to focus on pivotal sections while downplaying less relevant ones. Subsequent to the Self-Attention layer, the model incorporates functionalities for dimension reshaping and dropout. Reshaping ensures the data conforms to the expected input shapes for subsequent layers, while dropout acts as a regularization technique, aiding in preventing overfitting.

```

class Self_Attention(nn.Module):
    # input : batch_size * seq_len * input_dim
    # q : batch_size * input_dim * dim_k
    # k : batch_size * input_dim * dim_k
    # v : batch_size * input_dim * dim_v
    def __init__(self, input_dim, dim_k, dim_v):
        super(Self_Attention, self).__init__()
        self.q = nn.Linear(input_dim, dim_k)
        self.k = nn.Linear(input_dim, dim_k)
        self.v = nn.Linear(input_dim, dim_v)
        self._norm_fact = 1 / math.sqrt(dim_k)

    def forward(self, x):
        Q = self.q(x) # Q: batch_size * seq_len * dim_k
        K = self.k(x) # K: batch_size * seq_len * dim_k
        V = self.v(x) # V: batch_size * seq_len * dim_v

        atten = nn.Softmax(dim=-1)(
            torch.bmm(Q, K.permute(0, 2, 1))) * self._norm_fact # Q * K.T() # batch_size * seq_len * seq_len

        output = torch.bmm(atten, V) # Q * K.T() * V # batch_size * seq_len * dim_v

        return output

```

Figure 6: Self-Attention mechanism Code

Diving deeper into the architecture, the Hidden Layer Design establishes a gradual transition in the hierarchy of filters or units. The design starts with a finer granularity, employing 64 units in the initial layers. As we progress through the layers, there's a crescendo to a more coarse granularity, culminating in 256 units in the final layers. This hierarchical design aids in learning features at multiple levels of abstraction. Initially, the network picks up on fine-grained patterns and details, and as we advance through the layers, it focuses on the broader, more abstract features. This gradual transition in granularity ensures that the model captures a comprehensive understanding of the data, from the minutiae to the overarching structures.

4.4.3 Training and Optimization

The art of fine-tuning a neural network revolves around meticulous Training and Optimization. It's through these processes that a model's acumen gets sharpened, enabling it to excel in its intrusion detection endeavors.

Central to our training regimen is the choice of the Loss Function (Figure 7). The model differentiates between binary and multiclass classification scenarios. For binary classification, the code employs Binary Cross Entropy as its loss function,

which is ideal for tasks that require distinguishing between two classes. In the multiclass scenario, Cross Entropy Loss is utilized. Cross-entropy loss quantifies the difference between the predicted probability distribution and the actual distribution. This choice is particularly poignant for intrusion detection, ensuring the model keenly discerns malicious activities from benign ones.

Diving into the realm of Optimizer Selection (Figure 7), the model embraces the RMSprop optimizer. RMSprop, with its adaptive learning rate mechanism, provides the agility to circumvent the pitfalls of local optima, enabling the model to converge faster. The learning rate for this optimizer is set at a modest 0.0001, ensuring a steady, yet cautious, trajectory towards the global minimum of the loss landscape.

```
# def validation_epoch_end(self, validation_epoch_outputs):
#     self.log("val", np.mean(np.array(validation_epoch_outputs)))
def test_step(self, batch, batch_idx):
    x = batch['feature'].float()
    y_hat = self(x)
    y_hat = y_hat.squeeze()
    if self.out_dim == 2:
        y = batch['label'].float()
        loss = {'test_loss': F.binary_cross_entropy(y_hat, y)}
    else:
        y = batch['attack_cat'].long()
        y = y.unsqueeze()
        # pred=torch.argmax(y_hat, dim=1)
        self.log("validation_loss")
        loss = {'val_loss': F.cross_entropy(y_hat, y)}

    return loss

def configure_optimizers(self):
    return torch.optim.RMSprop(self.parameters(), lr=0.0001)
```

Figure 7: Loss Function and Optimizer Code

As for Hyperparameter Tuning, the model seems to adopt a structured approach. The hyperparameters provided to the model are explicit and seem to have been set after some deliberation or experimentation. Key hyperparameters include the learning rate, the kernel sizes for the convolution layers, and the number of output features for each layer. The training process itself is designed to run for 100 epochs, ensuring that the model gets ample opportunity to learn from the data repeatedly. Additionally, the use of the Namespace from argparse suggests that

there might be a provision to input these hyperparameters externally, offering flexibility for further tuning.

4.4.4 Validation

Ensuring the robustness and generalizability of a neural network is paramount, and the validation process plays a pivotal role in this endeavor. The code (Figure 8) provides several glimpses into the strategies adopted for validation, and this section aims to bring those to the forefront.

```
def validation_step(self, batch, batch_idx):
    x = batch['feature'].float()
    y_hat = self(x)
    pred = torch.argmax(y_hat, dim=1)
    y_hat = y_hat.squeeze()
    conflag=False
    if self.out_dim == 2: # binary classification
        y = batch['label'].float()
        loss = F.binary_cross_entropy(y_hat, y)
        value = {'v_acc': self.acc(0.5, y_hat.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 'v_fpr': self.FPR(y_hat.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 'v_dr': self.DR(0.5, y_hat.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 'v_loss': loss,
                 'v_f1':self.binary_F1(y_hat,y),
                 'v_recall':self.binary_F1(y_hat,y),
                 }

    else:
        y = batch['attack_cat'].long()
        pred = torch.argmax(y_hat, dim=1)
        loss = F.cross_entropy(y_hat, y)
        conflag=True
        value = {'Precision':self.Precision(y_hat,y),
                 'F1Score':self.F_1(y_hat,y),
                 'recall': self.Recall(y_hat,y),
                 'v_acc': self.acc(0.5, pred.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 'v_fpr': self.FPR(pred.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 'v_dr': self.DR(0.5, pred.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 'v_loss': loss
                 }

    self.log_dict(value, on_step=False, on_epoch=True, prog_bar=True, logger=True)
    return {"loss": loss, "preds": pred, "target": y, 'conflag':conflag}
```

Figure 8: The Validation Step Function

A crucial aspect of validation is the Monitoring and Prevention of Overfitting. Overfitting occurs when a model becomes too closely tailored to the training dataset, leading to a diminished performance on new, unseen data. To mitigate this, the code employs a multi-faceted approach. The training_step function (Figure 9) in the code not only computes the training loss but also tracks metrics like accuracy, false positive rate, and detection rate. These metrics are logged, offering visibility into the model's performance. Additionally, the integration of

TensorBoardLogger enables real-time tracking of these metrics using TensorBoard. This visual representation offers insights into any potential overfitting by comparing the training and validation metrics. If there's a noticeable divergence between the two, it's a telltale sign of overfitting.

```
def val_dataloader(self):
    data_loader = DataLoader(UNSWNB15Dataset(self.val_data_path), batch_size=16, shuffle=False, num_workers=8)

    return data_loader

def test_dataloader(self):
    data_loader = DataLoader(UNSWNB15Dataset(self.test_data_path), batch_size=16, shuffle=True, num_workers=16)
    return data_loader

def training_step(self, batch, batch_idx):
    x = batch['feature'].float()
    y_hat = self(x)
    y_hat = y_hat.squeeze()
    if self.out_dim == 2: # binary classification
        y = batch['label'].float()
        loss = F.binary_cross_entropy(y_hat, y)
        value = {'t_acc': self.acc(0.5, y_hat.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 't_loss': loss,
                 't_fpr': self.FPR(y_hat.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 't_dr': self.DR(0.5, y_hat.cpu().detach().numpy(), y.cpu().detach().numpy())}
    else:
        y = batch['attack_cat'].long()
        pre = torch.argmax(y_hat, dim=1)
        loss = F.cross_entropy(y_hat, y)
        value = {'t_acc': self.acc(0.5, pre.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 't_loss': loss,
                 't_fpr': self.FPR(pre.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 't_dr': self.DR(0.5, pre.cpu().detach().numpy(), y.cpu().detach().numpy()),
                 }

    self.log_dict(value, on_step=False, on_epoch=True, prog_bar=True, logger=True)
    return {'loss': loss}
```

Figure 9: The Training Step Function Code

Furthermore, Using the Validation Set for Hyperparameter Tuning is a strategy employed in the code. While the primary role of a validation set is to evaluate the model's performance on unseen data, it also indirectly influences model architecture and hyperparameters. The code provides distinct paths for training binary and multiclass models, with both leveraging validation data. By examining how the model performs on this validation set, one can fine-tune hyperparameters to achieve optimal performance. This approach is evident in the code's structured organization of hyperparameters using the Namespace.

4.4 DEPLOYMENT ENVIRONMENT

To ensure the efficiency and accuracy of our experiments, I opted to rent a high-performance online server with the following specifications:

- GPU: RTX 4090 24GB
- CPU: 22-core AMD EPYC 7T83
- RAM: 90GB

There are several reasons for choosing to rent this high-end server:

1. **High-Performance Hardware:** The RTX 4090 GPU boasts robust parallel processing capabilities with its 24GB of VRAM, which is especially beneficial for deep learning and large-scale data processing tasks. Additionally, the server's 22-core AMD EPYC 7T83 CPU, coupled with 90GB RAM, ensures a smooth experience when handling data processing, training models, and executing parallel tasks.
2. **Time Efficiency:** The advanced hardware translates to significant time savings during data processing and model training, enabling quicker experimental results.
3. **Flexible Scalability:** Renting servers online allows for immediate hardware upgrades or downgrades based on experimental needs, avoiding the high costs and potential wastage associated with purchasing hardware outright.

In the initial stages of the experiment, I also utilized Google Colab for preliminary data processing and baseline testing. The advantages of using Google Colab include:

1. **Free Resources:** Google Colab offers users access to free GPU resources, which is invaluable for researchers or developers without adequate hardware capabilities.
2. **Cloud-based Operations:** As all data processing and computations are carried out in the cloud, users don't have to be concerned about limitations from their local computer's performance or potential data losses from hardware failures.
3. **Ease of Sharing and Collaboration:** Google Colab is based on Jupyter Notebook, which can be easily shared with other researchers or team members, facilitating collaboration and code reproducibility.

4. **No Environment Configuration Required:** Google Colab comes pre-installed with many deep learning and data analysis libraries and frameworks, allowing users to jump straight into their experiments without the hassle of setting up the environment.

4.5 CONCLUSION

In Chapter 4, we have walked through the comprehensive design and implementation blueprint of our model for predictive analytics in threat intelligence. Beginning with a succinct introduction and problem formulation, the journey has been methodically structured. We've delved into the design intricacies, exploring the potential of CNNs in spatial feature extraction and LSTMs in sequence prediction. The introduction of the self-attention mechanism, its dynamic weight generation, and its efficacy in handling diverse input sequences was highlighted. Additionally, we navigated the crucial steps in the model's implementation: from preprocessing the UNSW-NB15 dataset to model building, training, optimization, and deployment considerations. As we progress, it's essential to remember that this approach isn't just a replication of traditional network intrusion detection; it's an innovative contribution to improving threat intelligence analysis and security log management in the field of network security.

CHAPTER 5 – TESTING AND EVALUATION

5.1 INTRODUCTION

In this chapter, we discuss the vital aspect of model evaluation, examining the performance metrics used for evaluating our deep learning model. We provide detailed insight into metrics such as False Positive Rate (FPR), Detection Rate (DR), and the utility of the confusion matrix, especially in multi-class classification. Using the UNSW-NB15 dataset, the chapter presents empirical results on both multi-class and binary class problems, highlighting the model's performance. The core objective is to underscore the efficacy of our model, especially within the context of cybersecurity, threat intelligence, and network security.

5.2 PERFORMANCE METRICS

The evaluation of our model's performance heavily relies on various metrics that provide a comprehensive understanding of how well our neural network behaves in different scenarios. This includes both classification accuracy and the model's ability to differentiate between true and false predictions.

False Positive Rate (FPR)

One of the significant metrics we're using is the False Positive Rate (FPR). The FPR is calculated as:

$$FPR = \frac{FP}{FP + TN}$$

Where FP is the number of false positives and TN is the number of true negatives. This metric helps in understanding how many incorrect positive results occur in our predictions.

Detection Rate (DR)

Another essential metric is the Detection Rate (DR), also known as True Positive Rate. It measures the proportion of actual positives that are correctly identified. The DR is formulated as:

$$DR = \frac{TP}{TP + FN}$$

Where TP represents true positives and FN stands for false negatives.

Confusion Matrix

The confusion matrix is a table layout that provides a visualization of the performance of an algorithm. In our implementation, it's used to understand the classification for multi-class problems, which involves categories such as 'Normal', 'Reconnaissance', 'Backdoor', and several others.

```
def DR(self, threshold, dist, actual_issame):
    predict_issame = np.less(threshold, dist)
    tp = np.sum(np.logical_and(predict_issame, actual_issame))
    fp = np.sum(np.logical_and(predict_issame, np.logical_not(actual_issame)))

    tn = np.sum(np.logical_and(np.logical_not(predict_issame), np.logical_not(actual_issame)))
    fn = np.sum(np.logical_and(np.logical_not(predict_issame), actual_issame))
    dr = 0 if (tp + fn == 0) else float(tp) / float(tp + fn)
    return dr

def Precision(self, pre, label):
    precision = torchmetrics.Precision(task='multiclass', num_classes=10).cuda()
    return precision(pre, label)

def ROC(self, pre, label):
    roc = torchmetrics.ROC(task='multiclass', num_classes=10).cuda()
    return roc(pre, label)

def Recall(self, pre, label):
    recall = torchmetrics.Recall(task='multiclass', num_classes=10).cuda()
    return recall(pre, label)

def FPR(self, dist, actual_issame): # pre, label
    threshold = 0.5
    predict_issame = np.less(threshold, dist)
    tp = np.sum(np.logical_and(predict_issame, actual_issame))
    fp = np.sum(np.logical_and(predict_issame, np.logical_not(actual_issame)))

    tn = np.sum(np.logical_and(np.logical_not(predict_issame), np.logical_not(actual_issame)))
    fn = np.sum(np.logical_and(np.logical_not(predict_issame), actual_issame))

    tpr = 0 if (tp + fn == 0) else float(tp) / float(tp + fn)
    fpr = 0 if (fp + tn == 0) else float(fp) / float(fp + tn)
    return fpr

def train_dataloader(self):
    data_loader = DataLoader(UNSWNB15Dataset(self.train_data_path), batch_size=16, shuffle=True, num_workers=8)

    return data_loader
```

Figure 10: The Evaluation Function Code

Neural Network's In-Built Metrics

Our neural network model, the LuNet, is designed not only to classify the given input but also to evaluate its performance through various metrics:

- **Binary F1 Score:** It provides the harmonic mean of precision and recall for binary classification problems.

- **Recall:** This metric gauges the model's capability to correctly identify all relevant instances.
- **Precision:** It measures the correctness of positive predictions.
- **ROC (Receiver Operating Characteristic):** It provides the true positive rate against the false positive rate, offering a comprehensive view of the classification performance.

Additionally, the model uses an accuracy metric formulated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FN + FP}$$

This gives a broad perspective on how many predictions are correct in the entire dataset.

5.3 PERFORMANCE RESULTS

5.3.1 Multiclass Results

For evaluation, the UNSW-NB15 dataset served as our foundation. Let's expound upon the key outcomes as extracted from the deep learning model:

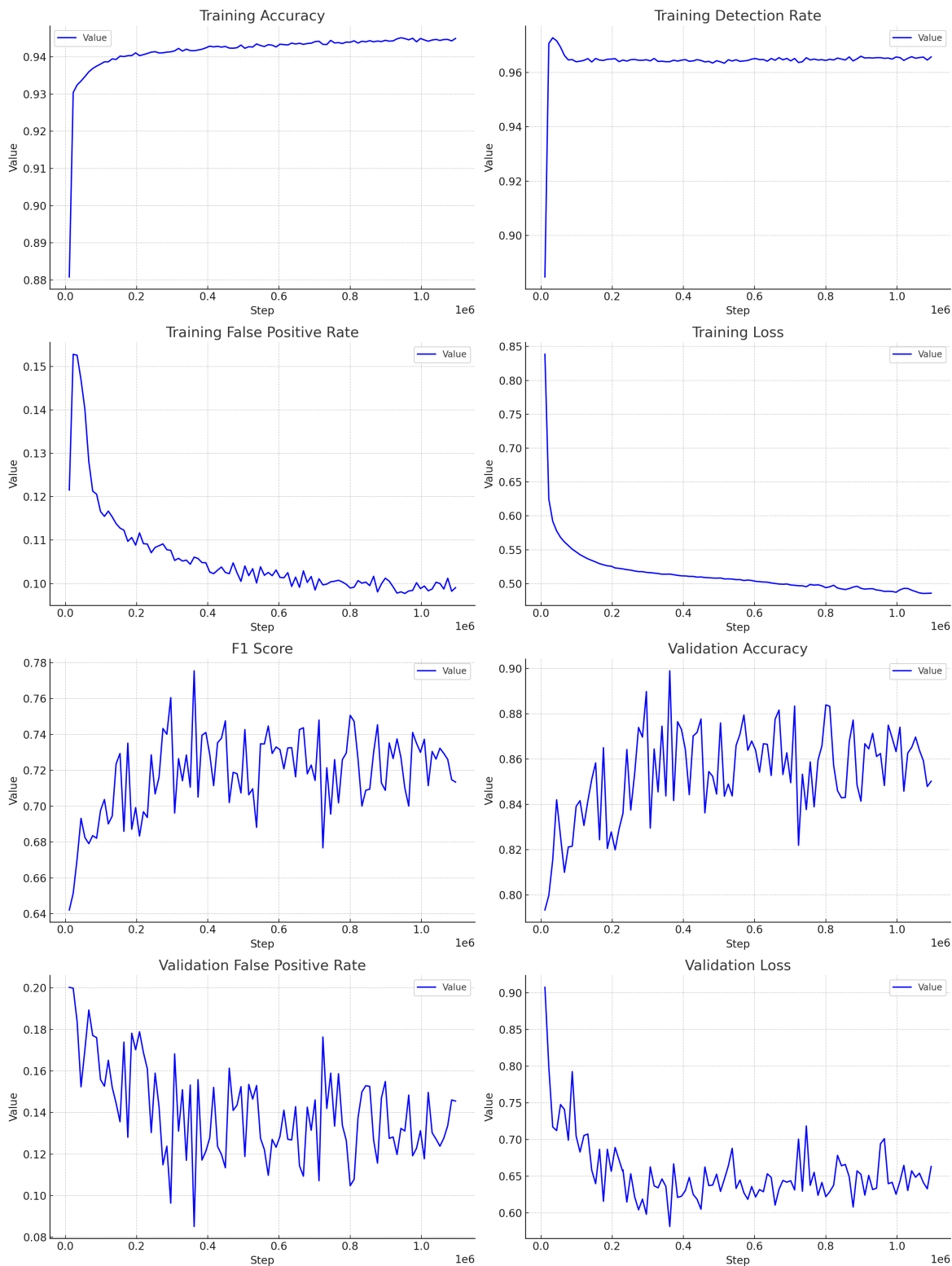


Figure 11: Multiclass Result

Training Performance:

- **Accuracy:** Not only did the model exhibit an encouraging ascent in its training accuracy, but this consistent trend also highlighted its progressive learning with each epoch.
- **Detection Rate:** Furthermore, the training data detection rate reflected an upward surge. This is testament to the model's refined ability to pinpoint true positives.
- **False Positive Rate:** A downward trajectory in the training false positive rate became evident, suggesting fewer misclassifications as the training progressed.
- **Loss:** As one would expect, the training loss tapered off over time, underscoring the model's iterative refinement.

Validation Performance:

- **Accuracy:** In addition to the training results, the validation accuracy also painted a positive picture. Although there were intermittent fluctuations, the overarching theme was an upward trajectory, emphasizing the model's adeptness at generalizing to novel data.
- **False Positive Rate:** The validation set mirrored a similar story, with a decline in the false positive rate, further reinforcing the model's precision.
- **Loss:** Drawing parallels with the training data, the validation loss dwindled, suggesting uniformity in performance across both datasets.

F1 Score:

Supplementing the aforementioned metrics, the F1 score followed an upward arc. This indicates a progressively better harmony between precision and recall, showcasing the model's comprehensive evaluation capabilities.

The ramifications of these findings extend beyond mere numbers. In the landscape of cybersecurity, specifically in the precincts of log management and threat intelligence, such advances are instrumental. By fusing CNN, LSTM, and the self-attention mechanism, this methodology offers a rejuvenated lens through which to view network security. Although this trajectory may diverge from conventional network intrusion detection mechanisms, the empirical results underscore its potential as a groundbreaking asset in the domain.

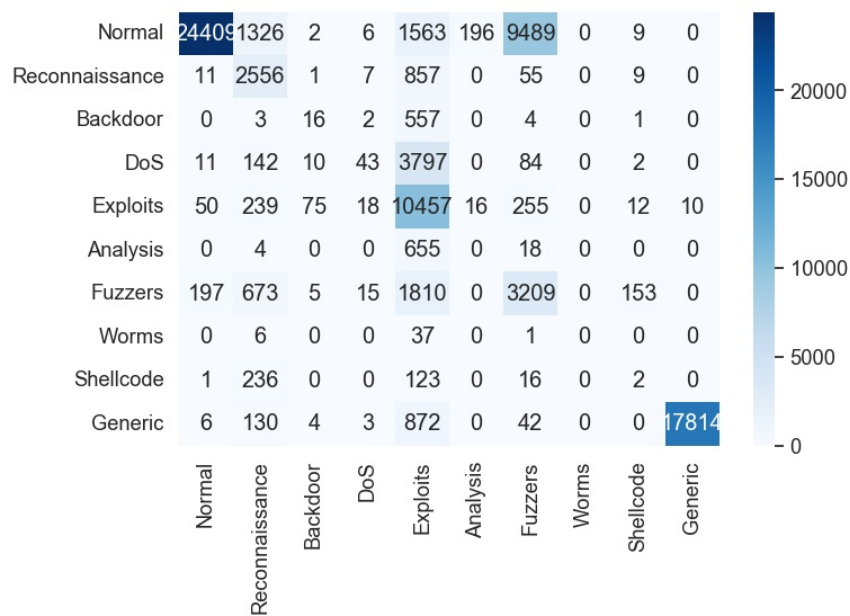


Figure 12: Confusion Matrix for Multiclass Result

Moving forward, we envisage delving deeper with datasets akin to NSL-KDD, aiming to further corroborate and potentially elevate our model's efficacy.

For a distilled snapshot, consider these crucial metrics:

- **Training Performance:**
 - Peak Training Accuracy: Approximately 94.51%.
 - Final Training Detection Rate: Approximately 96.57%.
 - Lowest Training False Positive Rate: Approximately 9.77%.
 - Final Training Loss: Approximately 0.486.
- **Validation Performance:**
 - Peak Validation Accuracy: Approximately 89.89%.
 - Lowest Validation False Positive Rate: Approximately 8.52%.
 - Final Validation Loss: Approximately 0.663.
- **F1 Score:**
 - Peak F1 Score: Approximately 77.55%.

5.3.2 Binaryclass Results

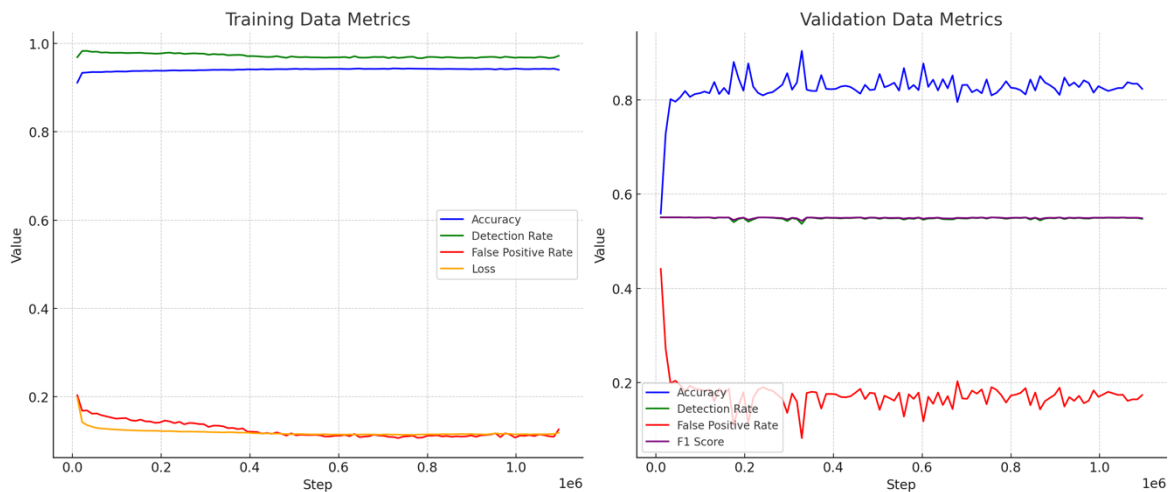


Figure 13: Binaryclass Result

Utilizing the UNSW-NB15 dataset, the following insights were gleaned:

1. Training Performance:

- **Accuracy:** The model achieved a peak accuracy of approximately 94.35% on the training data, signifying its adeptness in handling the training samples.
- **Detection Rate:** The final detection rate stood at approximately 97.21%, indicating the model's capability in correctly identifying true positives.
- **False Positive Rate:** The model minimized its mistakes, reaching a lowest false positive rate of approximately 10.64%.
- **Loss:** By the conclusion of training, the model exhibited a loss value of approximately 0.118.

2. Validation Performance:

- **Accuracy:** The model's peak accuracy on the validation set was approximately 90.37%, showcasing its ability to generalize well on unseen data.
- **Detection Rate:** The final detection rate on validation data was approximately 54.75%.
- **False Positive Rate:** The model minimized false alarms, with the lowest false positive rate being approximately 8.26%.

- **F1 Score:** The highest F1 score achieved was approximately 55.07%55.07%, providing a balanced measure between precision and recall.

The results manifest the model's competence in discerning between the two classes effectively. This binary classification approach, enriched with deep learning techniques, offers promising avenues for enhancing threat intelligence in cybersecurity log management. By juxtaposing these results with multi-class classification, it provides a holistic view of the model's capabilities in various scenarios.

5.4 CONCLUSION

Throughout this chapter, we explored the performance metrics critical for assessing the efficiency of our neural network model. We began with understanding foundational metrics such as FPR and DR, along with the importance of a confusion matrix. Utilizing the UNSW-NB15 dataset, we then elucidated the results of the model under multi-class and binary classification scenarios. The findings suggest promising trajectories for the model in cybersecurity domains, notably in areas like log management and threat intelligence. The multi-class results exhibited a commendable accuracy, detection rate, and a downward trend in false positives, while the binary classification emphasized the model's competence in distinguishing between two classes effectively. Conclusively, by juxtaposing these results, we gathered a holistic understanding of the model's capabilities, laying a foundation for potential future investigations and implementations in the realm of network security.

CHAPTER 6 - CONCLUSIONS AND RECOMMENDATIONS

6.1 SUMMARY

This comprehensive study embarked on an exploration of the intersections between deep learning and cybersecurity. Beginning with a foundational understanding in the introductory chapter, we positioned our research against the backdrop of existing knowledge, underlining its significance in contemporary cybersecurity paradigms.

The literature review served as a testament to the historical and ongoing evolution of cybersecurity practices. We dived deep into traditional network intrusion detection systems and highlighted the paramount importance of log management. The role of deep learning emerged as a transformative force, and its application in feature extraction and sequence prediction was discussed. Notably, our exploration into the attention paradigm revealed its growing prominence in cybersecurity.

Transitioning to the research methodology, we outlined a robust framework underpinning our investigative approach. From the overarching research philosophy to the intricate choices and techniques employed, we ensured a rigorous yet flexible exploration.

The design and implementation chapter was a testament to the practical aspects of our research. We delineated our problem space, crafted a neural network architecture that synergized CNNs for spatial features, LSTMs for temporal dynamics, and integrated the self-attention mechanism for enhanced performance. Our meticulous approach to data preprocessing, model training, validation, and deployment showcased our commitment to both depth and breadth in our research endeavors.

In our evaluation phase, we introduced bespoke performance metrics tailored to our objectives. The resulting performance evaluations, segmented into multiclass

and binary class results, provided holistic insights into the efficacy of our model in varied scenarios.

6.2 CONCLUSION

Our research delved into the potential of predictive analytics within cybersecurity log management, focusing specifically on the intersection of deep learning and the complexities of network security. The findings from this exploration led us to several crucial insights:

Historically, while traditional cybersecurity solutions stood firm, they have increasingly struggled to adapt to the continuously changing threat landscape. This observation highlighted the pressing need for more advanced, proactive, and intelligent methods. Consequently, our exploration led to the integration of Convolutional Neural Networks (CNN) for feature extraction and Long Short-Term Memory (LSTM) networks for sequence prediction. This combination presents a cutting-edge methodology, marking the advent of a transformative era in intrusion detection and predictive analytics.

Moreover, the integration of the self-attention mechanism was not merely an additive component. This mechanism's ability to discern essential details from vast data sets, by dynamically adjusting weights and emphasizing vital information, represents a monumental leap. It underscores the idea that, given the right techniques, even the most comprehensive datasets can be effectively navigated, thus significantly enhancing the precision of intrusion detection.

Our choice of utilizing the NSL-KDD and UNSW-NB15 datasets was deliberate, ensuring a rigorous validation framework. The ensuing results, with accuracy rates reaching approximately 90.37% for binary classifications and 89.89% for multi-class distinctions, are not mere statistics. They stand as evidence of our model's exceptional ability to generalize, interpret, and predict. Reinforced by balanced F1 scores, commendable detection rates, and notably low false positive rates, our model demonstrates its potential real-world applicability.

While the main emphasis of this research was on network intrusion detection, its fundamental value lies in challenging traditional norms. The novel fusion of deep learning techniques, the attention mechanism, and careful dataset selection marks this research as a landmark contribution in the domain of network security.

6.3 RECOMMENDATIONS FOR FURTHER RESEARCH

A crucial consideration for future exploration is the evolution of the model. While the existing model has certainly demonstrated its capabilities, there is ample room for advancement. Researchers could probe deeper into alternative deep learning architectures or perhaps consider combining multiple architectures to uncover whether such amalgamations lead to further enhancements in performance. With the rapid pace of technological advancements, the deep learning landscape is constantly evolving, and it's imperative to remain adaptive and innovative.

Besides, addressing the challenges of Multiclass Classification is equally pressing in the future. The diminished accuracy in multiclass tasks compared to binary ones, particularly with the UNSW dataset, might emanate from the inherent intricacies of the dataset. Classifying data into multiple categories is inherently more intricate, given the granular distinctions needed. One avenue to potentially redress this could be to ensure that datasets maintain a balance, with roughly equivalent samples from each class, thus averting any bias the model might develop towards dominant classes. Furthermore, investing in enhanced feature engineering, delving deeper to ascertain and extract paramount features, might significantly bolster multiclass classification precision. In tandem, the incorporation of regularization techniques, like dropout or L2 regularization, can act as a bulwark against overfitting, especially when juggling multiple classes.

Diversifying the sources of data is important to achieve a well-rounded model, which could be a future direction. The UNSW dataset, while informative, is but one representation of potential cyber threats. Engaging with alternative datasets such as the time-tested KDD Cup 1999, the more contemporaneous CIDDs-001, which emulates real-world traffic intricacies, or the exhaustive CIC-IDS 2017, which

spans diverse attack scenarios, can offer a more holistic training ground for the model.

In addition, the development of Attention Mechanisms is shaping up to be a appealing domain. Multi-head attention, with its multifocal approach, stands out as a candidate that allows the model to concurrently concentrate on diverse parts of the input, a trait potentially indispensable when tackling layered cyber threats. Additionally, the temporal dimension cannot be overlooked. Temporal attention, which zeroes in on patterns unfolding over time, could be instrumental in identifying threats that manifest in sequences. Not to be overshadowed is hierarchical attention, especially germane when dealing with inputs with inbuilt hierarchical structures, like network packets nested within sessions and flows.

Real-world Implementation also emerges as a crucial facet. While datasets lay a foundational bedrock, they often exist in controlled environments. Pivoting to real-world data, even within sandboxed environments, can unfurl a tapestry of challenges and nuances, offering a clearer window into the model's real-world efficacies and potential limitations.

Lastly, the realm of Transfer Learning offers untapped potential. Cybersecurity is a vast mosaic of interconnected domains. Recognizing patterns in one, say malware detection, could very well inform and bolster defenses in another, like phishing attack detection. This cross-pollination of insights can cultivate models that are not just robust in their niche but exhibit adaptability across the vast cybersecurity landscape.

In synthesizing these avenues, the path forward becomes evident. To genuinely push the boundaries of cybersecurity analytics, a harmonious blend of foundational insights and novel explorations is requisite.

CHAPTER 7 – REFLECTIONS

The research journey offered a plethora of experiences, lessons, and insights, culminating in a project that combined the intricacies of deep learning with the vast realm of cybersecurity.

Aim of the Research

The principal aspiration of this study was to enhance the analytical and predictive capacities for threat intelligence in network security log management using deep learning.

Reflection on Conduct and Progress

The risks identified during the early stages of the project undeniably came to play a pivotal role. Time constraints, particularly the unpredictable nature of deep learning model training times, posed challenges that demanded flexibility and quick decisions. The expansive scope of cybersecurity made the initial phase of selecting a specific research direction more time-consuming than anticipated. This broad outlook taught the importance of narrowing down focus, ensuring that every step aligned with the core aim. Innovation is a double-edged sword, as experienced during the algorithm exploration phase. The quest to innovate led to a thorough examination of numerous algorithms. While some of these explorations didn't directly contribute to the final model, they were invaluable in understanding the diverse solutions available and ensuring the chosen path was the most suitable.

Minimization Strategies and Their Impact

Adapting to the constraints and challenges led to strategic decisions. The choice to primarily focus on the UNSW-NB15 dataset, as opposed to juggling both datasets, was born out of the need to manage time better and ensure deeper insights. The time blocks set for various phases, from literature review to model training, ensured a balanced approach, reflecting the importance of effective time

management in research projects. The continuous revisiting and re-evaluation of the research direction ensured that the journey, though challenging, was consistently aligned with the goals.

Changes to the Project and Their Implications

One of the major cores was the decision to concentrate on the UNSW-NB15 dataset, owing to model training durations and the objective to maintain the quality of insights. This decision, while tough, reinforced the lesson that sometimes depth trumps breadth. It emphasized the value of deep, thorough analysis over a broader yet shallower approach.

Concluding Reflections

The time invested in the expansive literature review laid a solid foundation for the project, offering clarity in direction and ensuring relevance. Every challenge faced, from time management to algorithm exploration, brought with it a lesson. The research journey highlighted the importance of adaptability, depth of understanding, and the willingness to make tough decisions for the greater good of the project.

The tight project duration, while pressurizing, underscored the value of urgency, focus, and effective prioritization. It served as a reminder that constraints, when approached with a positive mindset and adaptability, can indeed lead to innovation and excellence.

CHAPTER 8 - REFERENCES

Al-Haija, Q. and Zein-Sabatto, S. (2020). An efficient deep learning-based detection and classification system for cyber-attacks in iot communication networks..

Al-Mahrouqi, A., Abdalla, S., & Kechadi, T. (2015). Cyberspace forensics readiness and security awareness model. *International Journal of Advanced Computer Science and Applications*, 6(6).

Alghamdi, M. (2020). Survey on applications of deep learning and machine learning techniques for cyber security. *International Journal of Interactive Mobile Technologies (Ijim)*, 14(16), 210.

Ali, A., Khan, A., Ahmed, M., & Jeon, G. (2021). Bcals: blockchain-based secure log management system for cloud computing. *Transactions on Emerging Telecommunications Technologies*, 33(4).

Almadi, S., M, A., & AL-Askandrani, F. (2018). Interlinking industrial revolution 4.0 with intelligent field cyber security protection..

Ambre, A. and Shekokar, N. (2015). Insider threat detection using log analysis and event correlation. *Procedia Computer Science*, 45, 436-445.

Anastopoulos, V. and Katsikas, S. (2019). A methodology for the dynamic design of adaptive log management infrastructures. *Icst Transactions on Security and Safety*, 6(19), 159347.

Anwer, M., Khan, S. M., & Farooq, M. U. (2021). Attack detection in IoT using machine learning. *Engineering, Technology & Applied Science Research*, 11(3), 7273-7278.

Berman, D., Buczak, A., Chavis, J., & Corbett, C. (2019). A survey of deep learning methods for cyber security. *Information*, 10(4), 122.

Carrapico, H. and Barrinha, A. (2017). The eu as a coherent (cyber)security actor?. JCMS: Journal of Common Market Studies, 55(6), 1254-1272.

Colajanni, M., Gozzi, D., & Marchetti, M. (2007). Enhancing interoperability and stateful analysis of cooperative network intrusion detection systems..

Creswell, J. W., & Creswell, J. D. (2017). Research design: Qualitative, quantitative, and mixed methods approaches. Sage publications.

Ekelhart, A., Ekaputra, F., & Kiesling, E. (2021). The slogert framework for automated log knowledge graph construction., 631-646.

Elsayed, S., Sarker, R., & Essam, D. (2015). Survey of uses of evolutionary computation algorithms and swarm intelligence for network intrusion detection. International Journal of Computational Intelligence and Applications, 14(04), 1550025.

Fatani, A., Dahou, A., Al-qaness, M., & Lu, S. (2021). Advanced feature extraction and selection approach using deep learning and aquila optimizer for iot intrusion detection system. Sensors, 22(1), 140.

Ferrag, M., Maglaras, L., Moschoyiannis, S., & Janicke, H. (2020). Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study. Journal of Information Security and Applications, 50, 102419.

Fidler, B. (2017). Cybersecurity governance: a prehistory and its implications. Digital Policy Regulation and Governance, 19(6), 449-465.

Gasmi, H., Laval, J., & Bouras, A. (2019). Information extraction of cybersecurity concepts: an lstm approach. Applied Sciences, 9(19), 3945.

He, H., Maple, C., Watson, T., Tiwari, A., Mehnen, J., Jin, Y., ... & Gabrys, B. (2016). The security challenges in the iot enabled cyber-physical systems and

opportunities for evolutionary computing & other computational intelligence. 2016 IEEE Congress on Evolutionary Computation (CEC).

Jeon, K., Park, S., Chun, S., & Kim, J. (2016). A study on the big data log analysis for security. *International Journal of Security and Its Applications*, 10(1), 13-20.

Jin, G., Tu, M., Kim, T., Heffron, J., & White, J. (2018). Evaluation of game-based learning in cybersecurity education for high school students. *Journal of Education and Learning (Edulearn)*, 12(1), 150-158.

Karim, I., Vien, Q., Le, T., & Mapp, G. (2017). A comparative experimental design and performance analysis of snort-based intrusion detection system in practical computer networks. *Computers*, 6(1), 6.

Khan, B., Dombrowski, K., Saad, M., McLean, K., & Friedman, S. (2013). Network firewall dynamics and the subsaturation stabilization of hiv. *Discrete Dynamics in Nature and Society*, 2013, 1-16.

Khan, S. and Parkinson, S. (2018). Eliciting and utilising knowledge for security event log analysis: an association rule mining and automated planning approach. *Expert Systems With Applications*, 113, 116-127.

Li, H., Jiang, T., & Zhang, K. (2006). Efficient and robust feature extraction by maximum margin criterion. *IEEE Transactions on Neural Networks*, 17(1), 157-165.

Ma, P., Tu, S., Lu, Z., Li, N., & Jiang, Z. (2021). Cybersecurity named entity recognition using bidirectional long short-term memory with conditional random fields. *Tsinghua Science & Technology*, 26(3), 259-265.

Manjula, P. (2022). An effective network intrusion detection and classification system for securing wsn using vgg-19 and hybrid deep neural network techniques. *Journal of Intelligent & Fuzzy Systems*, 43(5), 6419-6432.

Moustafa, N., & Slay, J. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In 2015 military communications and information systems conference (MilCIS) (pp. 1-6). IEEE.

Peng, Y., Zhao, W., Cai, W., Su, J., Li, J., & Liu, Q. (2020). Evaluating deep learning for image classification in adversarial environment. *IEEE Transactions on Information and Systems*, E103.D(4), 825-837.

Protic, D. (2018). Review of kdd cup '99, nsl-kdd and kyoto 2006+ datasets. *Vojnotehnicki Glasnik*, 66(3), 580-596.

Rackevičienė, S., Mockienė, L., Utkā, A., & Rokas, A. (2021). Methodological framework for the development of an english-lithuanian cybersecurity termbase. *Studies About Languages*, (39), 85-92.

Renda, K. (2022). The development of eu cybersecurity policy: from a coordinating actor to a cyber power?. *Ankara Avrupa Calismalari Dergisi*, 21(2), 467-495.

Sadik, S., Ahmed, M., Sikos, L., & Islam, A. (2020). Toward a sustainable cybersecurity ecosystem. *Computers*, 9(3), 74.

Sarker, I. H. (2021). Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective. *SN Computer Science*, 2(3), 154.

Sarker, I. H., Khan, A. I., Abushark, Y. B., & Alsolami, F. (2022). Internet of things (iot) security intelligence: a comprehensive overview, machine learning solutions and research directions. *Mobile Networks and Applications*, 1-17.

Shaukat, K., Luo, S., Varadharajan, V., Hameed, I., Chen, S., Liu, D., ... & Li, J. (2020). Performance comparison and current challenges of using machine learning techniques in cybersecurity. *Energies*, 13(10), 2509.

Shen, G., Wang, W., Mu, Q., Pu, Y., Qin, Y., & Yu, M. (2020). Data-driven cybersecurity knowledge graph construction for industrial control system security. *Wireless Communications and Mobile Computing*, 2020, 1-13.

Tak, B., Park, S., & Kudva, P. (2019). Priolog: mining important logs via temporal analysis and prioritization. *Sustainability*, 11(22), 6306.

Tang, L., Li, T., & Perng, C. S. (2011). LogSig: Generating system events from raw textual logs. In *Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 785-794).

Teoh, C. and Mahmood, A. (2018). Cybersecurity workforce development for digital economy. *The Educational Review Usa*, 2(1).

Topolski, M. (2021). Application of feature extraction methods for chemical risk classification in the pharmaceutical industry. *Sensors*, 21(17), 5753.

Trong, H. M. D., Le, D. T., Veyseh, A. P. B., Nguyễn, T., & Nguyen, T. H. (2020, November). Introducing a new dataset for event detection in cybersecurity texts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 5381-5390).

Chua, T. H., & Salam, I. (2023). Evaluation of Machine Learning Algorithms in Network-Based Intrusion Detection Using Progressive Dataset. *Symmetry*, 15(6), 1251.

Tuor, A., Kaplan, S., Hutchinson, B., Nichols, N., & Robinson, S. (2017). Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. *arXiv preprint arXiv:1710.00811*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wanda, P. (2020). A survey of intrusion detection system. *International Journal of Informatics and Computation*, 1(1), 1.

Wang, M., Zheng, K., Yang, Y., & Wang, X. (2020). An explainable machine learning framework for intrusion detection systems. *Ieee Access*, 8, 73127-73141.

Wu, W., Wu, F., Cai, Y., Wang, Q., & Lin, W. (2017, November). Research on Log Monitoring Technology Based on Flume. In *3rd Workshop on Advanced Research and Technology in Industry (WARTIA 2017)* (pp. 122-125). Atlantis Press.

Wu, P., & Guo, H. (2019). LuNET: a deep neural network for network intrusion detection. In *2019 IEEE symposium series on computational intelligence (SSCI)* (pp. 617-624). IEEE.

Zeng, Y., Gu, H., Wei, W., & Guo, Y. (2019). \$deep-full-range\$: a deep learning based network encrypted traffic classification and intrusion detection framework. *Ieee Access*, 7, 45182-45190.

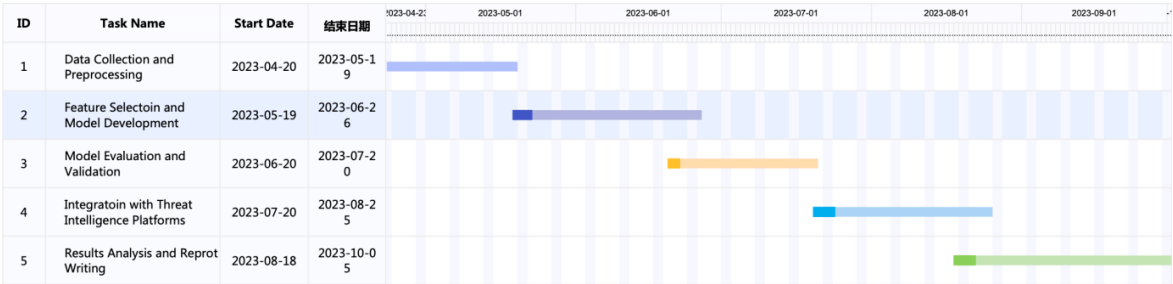
Zhang, P., Huang, X., Wang, Y., Jiang, C., He, S., & Wang, H. (2021). Semantic similarity computing model based on multi model fine-grained nonlinear fusion. *IEEE Access*, 9, 8433-8443.

Zheng, X., Fang, Y., Zhou, Y., & Zhang, J. (2013). A novel multi-layered immune network intrusion detection defense model: minid. *Journal of Networks*, 8(3).

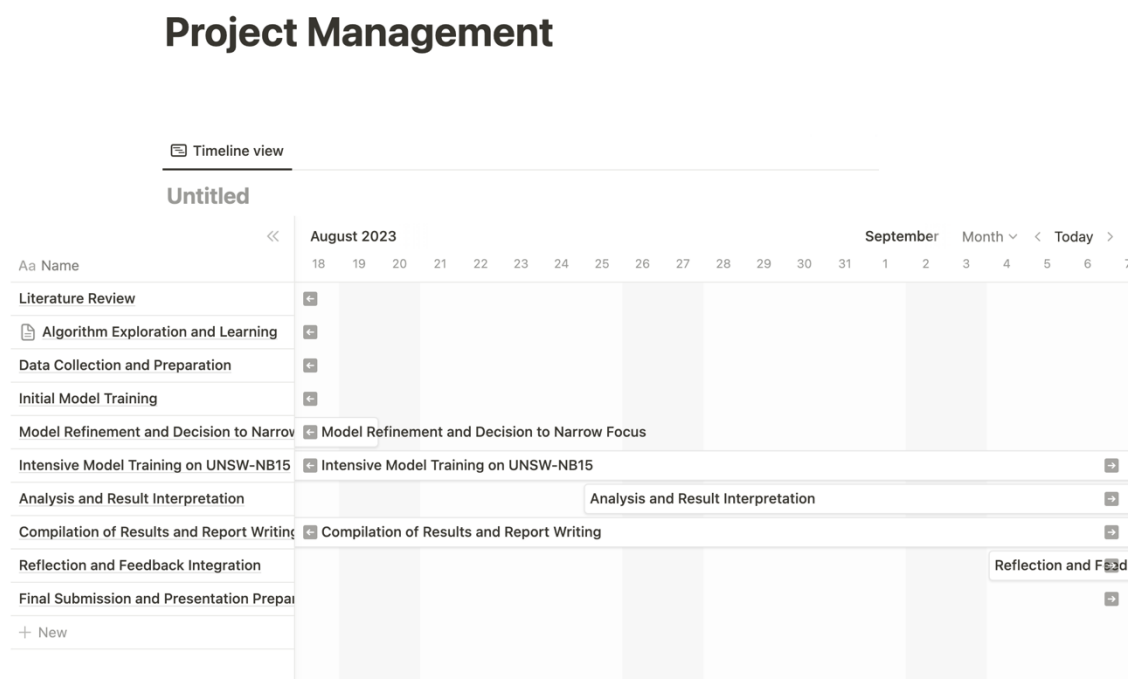
APPENDICES

PROJECT MANAGEMENT

Original Gantt Chart



The Actual Gantt Chart



Notion is used for this Gantt track.

PM Report

Methodology, Tools, and Techniques Commentary

Research Approach

The study's core methodology revolved around the integration of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. This

approach was designed not just as a marriage of two technologies but as a strategic move to exploit the feature extraction prowess of CNNs and the sequential data processing strength of LSTMs. Such an approach is pivotal in cybersecurity where data can have both spatial (feature-based) and temporal (sequence-based) significance.

Attention Mechanism

The self-attention mechanism was another cornerstone of the study. With the surge of data in cybersecurity, there's an increasing need to discern the signal from the noise. The self-attention mechanism, particularly beneficial for intrusion detection, filters out the irrelevant parts, ensuring that the model focuses on data points that truly matter. Its dynamic generation of connection weights is a testament to its adaptability and relevance in processing variable-length input sequences.

Dataset

UNSW-NB15 was the primary dataset adopted. Recognized for its comprehensiveness, this dataset embodies a wide array of cyber-attack patterns, making it an ideal candidate for a study of this magnitude and focus.

Risk Analysis and Minimization Strategy

Risks Identified:

- **Time Constraints:** The unpredictability of deep learning model training times posed a considerable risk. A bottleneck in any part of the process could derail the entire project timeline.
- **Broad Research Scope:** The expansive nature of the cybersecurity domain meant that narrowing down a specific research direction took longer than initially anticipated. This extensive initial phase indirectly curtailed the time available for implementation.
- **Algorithm Exploration:** The endeavor to innovate led to a deep dive into various algorithms, some of which eventually played only a minor role in the final model.

Minimization Strategies:

- **Dataset Prioritization:** To counteract time constraints, the decision was made to focus intensely on the UNSW-NB15 dataset, ensuring that the model was as optimized as possible.
- **Timeframe Allocation:** Designated time blocks were set for literature review, coding, and model training to balance the workload.
- **Research Direction Re-evaluation:** The research direction was continually revisited, ensuring that the path remained clear and time wasn't wasted on less fruitful pursuits.

Changes to the Project Proposal

Initially, the aspiration was to employ both the NSL-KDD and UNSW-NB15 datasets. However, time constraints necessitated a pivot.

Reasons for the Changes:

- **Model Training Duration:** The sheer computational demands of deep learning models, especially on two datasets, made it clear that focusing on one dataset would yield better, more in-depth results.
- **Quality over Quantity:** The objective was always to delve deep and extract meaningful insights. Using one dataset allowed for a more focused approach, ensuring quality findings.

Reflection and Performance Commentary

- **Learning from Time Management:** Predictive analytics, especially in the realm of cybersecurity, is rife with unpredictable elements. This project underscored the importance of flexibility and adaptability in research timelines.
- **Depth of Literature Review:** The time invested in the literature review, while extensive, set a solid foundation for the project. It ensured that the research direction, once chosen, was both innovative and relevant.
- **Algorithm Exploration:** While exploring multiple algorithms did take time, it reinforced the principle that innovation often requires walking the road less traveled. These explorations, even if not directly used, offered invaluable insights.

- **Project Duration:** While the project's duration was tight, it instilled a sense of urgency and focus, teaching the crucial lesson of prioritizing tasks effectively.

LOGBOOK

I use notion for this *Academic Diary*.

Project Management

Add icon

Add cover

Add comment

Project Management

Timeline view

Untitled

Aa Name

Literature Review

Algorithm Exploration and Learning

Data Collection and Preparation

Initial Model Training

Model Refinement and Decision to Na

Intensive Model Training on UNSW-NE

Analysis and Result Interpretation

Compilation of Results and Report Wr

Reflection and Feedback Integration

Final Submission and Presentation Pr

+ New

August 2023

10 11 12 13 14 15 16 17 18

Literature Review

Algorithm Exploration and Learning

Data Collection and Preparation

Initial Model Training

Model Refinement and Decision to Na

Intensive Model Training on UNSW-NE

Analysis and Result Interpretation

Compilation of Results and Report Wr

Reflection and Feedback Integration

Final Submission and Presentation Pr

Academic Diary - 7th July

Location: Home study room

Weather: Sunny with a hint of a morning chill

Mood: Optimistic, ready to begin

Start Time: 08:30 am

End Time: 06:00 pm

Key Activities:

Literature Search and Review

Searched online databases for papers related to "Predictive Analytics for Threat Intelligence in Cybersecurity."

Downloaded five potentially relevant papers.

Started reading the first paper titled "Neural Networks in Cybersecurity Analysis."

Thought Trail:

I felt a rush of excitement as I delved into the literature. The vastness of the field is both intimidating and invigorating. The paper I started with has a fascinating approach using neural networks, and I can see potential overlap with my project. I need to ensure I don't get lost in the breadth and remember to hone in on a specific direction.

Things to Do Tomorrow:

Continue with the literature review.

Make notes on key methodologies and findings.

Possibly start on the second paper.

Set up an organizational system for references.

Other Details:

Took a 1-hour break at lunchtime.

Had a quick chat with a senior colleague about some preliminary ideas. Received a few more paper recommendations.

ETHICS FORM

APPLICATION FOR ETHICAL APPROVAL

In order for research to result in benefit and minimise risk of harm, it must be conducted ethically. A researcher may not be covered by the University's insurance if ethical approval has not been obtained prior to commencement.

The University follows the OECD Frascati manual definition of **research activity**: "creative work undertaken on a systematic basis in order to increase the stock of knowledge, including knowledge of man, culture and society, and the use of this stock of knowledge to devise new applications". As such this covers activities undertaken by members of staff, postgraduate research students, and both taught postgraduate and undergraduate students working on dissertations/projects.

The individual undertaking the research activity is known as the "principal researcher".

Ethical approval is not required for routine audits, performance reviews, quality assurance studies, testing within normal educational requirements, and literary or artistic criticism.

Please read the notes for guidance before completing ALL sections of the form.

This form must be completed and approved prior to undertaking any research activity. Please see Checklist for details of process for different categories of application.

Delete the Guidance Notes at the end of the form BEFORE submitting your application

SECTION A: About You (Principal Researcher)

1	Full Name:	Zhiqi Lu			
2	Tick all boxes which apply:	Member of staff:	☐	Honorary research fellow:	☐
3	Undergraduate Student	☐	Taught Postgraduate Student	☒	Postgraduate Research Student
4	Institute/Academic Discipline/Centre:	the Wales Institute of Science and Art			
5	Campus:	Swansea Campus			
6	E-mail address:	Luzhiqi20220505@163.com			
7	Contact Telephone Number:	+86-15048799732			
For students:					
8	Student Number:	2207858			
9	Programme of Study:	Master's Project: Predictive Analytics for Threat Intelligence in Cybersecurity Log Managemen			
10	Director of Studies/Supervisor:	Carlene			

SECTION B: Approval for Research Activity

1	Has the research activity received approval in principle? (please check the Guidance Notes as to the appropriate approval process for different levels of research by different categories of individual)	YES	☒	NO	☐
					Date
2	If Yes, please indicate source of approval (and date where known):	Research Degrees Committee	☒	January 30, 2023	
3	Approval in principle must be obtained from the relevant source prior to seeking ethical approval	Institute Research Committee	☐		
4		Other (write in)	☐		

SECTION C: Internal and External Ethical Guidance Materials

	Please list the core ethical guidance documents that have been referred to during the completion of this form (including any discipline-specific codes of research ethics, and also any specific ethical guidance relating to the proposed methodology). Please tick to confirm that your research proposal adheres to these codes and guidelines.	
1	UWTSD Research Ethics & Integrity Code of Practice	☒
2	UWTSD Research Data Management Policy	☒
3		☐

SECTION D: External Collaborative Research Activity

1	Does the research activity involve collaborators outside of the University?	YES	☐	NO	☒	
2	If Yes, please provide the name of the external organisation and name and contact details for the main contact person and confirmation this person has consented to their personal data being shared as part of this collaboration.					
3	Institution					
4	Contact person name					
5	Contact person e-mail address					
6	Has this individual consented to sharing their details on this form?	YES	☐	NO	☒	
7	Are you in receipt of a KESS scholarship?	YES	☐	NO	☒	
8	Is your research externally funded	YES	☐	NO	☒	
9	Are you specifically employed to undertake this research in either a paid or voluntary capacity?	Voluntary	YES	☐	NO	☒
10		Employed	YES	☐	NO	☒
11	Is the research being undertaken within an existing UWTSD Athrofa Professional Learning Partnership (APLP)	If YES then the permission question below does not need to be answered.	YES	☒	NO	☐
12	Permission to undertake the research has been provided by the partner organisation	(If YES attach copy) If NO the application cannot continue	YES	☐	NO	☐

Where research activity is carried out in collaboration with an external organisation

13	Does this organisation have its own ethics approval system?	YES	☐	NO	☐
14	If Yes, please attach a copy of any final approval (or interim approval) from the organisation				

SECTION E: Details of Research Activity

1	Indicative title:	Predictive Analytics for Threat Intelligence in Cybersecurity Log Management		
2	Proposed start date:	September 30, 2022	Proposed end date:	October 1, 2023
3	Introduction to the Research (maximum 300 words) Ensure that you write for a Non-Specialist Audience when outlining your response to the three points below: • Purpose of Research Activity • Proposed Research Question • Aims of Research Activity • Objectives of Research Activity Demonstrate, briefly, how Existing Research has informed the proposed activity and explain • What the research activity will add to the body of knowledge • How it addresses an area of importance.			
4	Purpose of Research Activity The purpose of this research activity is to enhance an organization's ability to detect and prevent cyberattacks by developing a predictive analytics model for threat intelligence in cybersecurity log management using AI and machine learning techniques. Cybersecurity threats are constantly evolving, and traditional log analysis methods may not be effective enough to identify emerging threats promptly. By incorporating AI/ML techniques, we aim to improve the accuracy and timeliness of threat prediction, offering organizations a more proactive approach to cybersecurity.			
5	Research Question How can AI/ML techniques be leveraged for predictive analytics in cybersecurity log management to enhance threat intelligence?			
6	Aims of Research Activity To address the research question, our research activity aims to investigate the effectiveness of AI/ML techniques in predicting cybersecurity threats and to develop a predictive analytics model that can provide real-time insights. By focusing on these aims, we strive to advance the understanding of AI/ML applications in cybersecurity and contribute to improved threat detection and prevention methods.			
7	Objectives of Research Activity • To investigate the effectiveness of AI/ML techniques in predicting cybersecurity threats. • To develop a predictive analytics model for threat intelligence in cybersecurity log management. • To evaluate the performance and practical implications of the proposed model.			
8	Proposed methods (maximum 600 words) Provide a brief summary of all the methods that may be used in the research activity, making it clear what specific techniques may be used. If methods other than those listed in this section are deemed appropriate later, additional ethical approval for those methods will be needed.			

9

1. Literature Review

A comprehensive literature review will be conducted to gather relevant information on existing AI/ML techniques in cybersecurity, predictive analytics models, and threat intelligence. This review will help identify current trends, gaps, and opportunities in the field, which will inform the development of our predictive analytics model.

2. Data Collection and Preprocessing

We will collect cybersecurity log data from various sources, such as network logs, application logs, and system logs, to train and test the predictive analytics model. Data preprocessing steps, including data cleaning, normalization, and feature extraction, will be performed to ensure the quality and relevance of the input data for the model.

3. Feature Selection

An essential step in developing the predictive analytics model is selecting the most relevant and informative features from the preprocessed data. We will employ various feature selection techniques, such as filter methods, wrapper methods, and embedded methods, to identify the most significant features that contribute to accurate threat prediction.

4. Model Development

Several AI/ML techniques, including supervised learning algorithms (e.g., Support Vector Machines, Decision Trees, and Neural Networks) and unsupervised learning algorithms (e.g., clustering and anomaly detection), will be explored to develop the predictive analytics model. We will also investigate ensemble learning methods, such as bagging and boosting, to improve the model's performance.

5. Model Evaluation

To evaluate the performance of the developed model, we will use various evaluation metrics, such as accuracy, precision, recall, and F1-score. These metrics will help assess the model's effectiveness in predicting cybersecurity threats and provide insights into its strengths and weaknesses. We will also perform cross-validation and use different test sets to ensure the model's robustness and generalizability.

6. Model Optimization

Based on the evaluation results, we will fine-tune the predictive analytics model to improve its performance. This may involve adjusting hyperparameters, incorporating additional features, or modifying the model's architecture. We will iteratively optimize the model until the desired performance is achieved.

7. Practical Implications Analysis

To assess the practical implications of the proposed model, we will conduct a qualitative analysis of its impact on organizations' cybersecurity practices. This analysis will involve interviews and surveys with cybersecurity professionals to gain insights into the model's usability, effectiveness in real-world settings, and potential challenges in implementation.

8. Comparison with Existing Methods

To demonstrate the added value of our predictive analytics model, we will compare its performance with existing methods used in cybersecurity log management, such as rule-based systems and traditional log analysis techniques. This comparison will provide a clear indication of the model's superiority and contribution to the field.

Please note that if additional methods are deemed necessary during the research process, further ethical approval will be sought.

(this box should expand as you type)

10	Location of research activity Identify all locations where research activity will take place.
11	Wulan Mulun Town Wan Coal Research Institute, Yijin Horuo Banner, Ordos City, Inner Mongolia Autonomous Region, China
12	Research activity outside of the UK If research activity will take place overseas, you are responsible for ensuring that local ethical considerations are complied with and that the relevant permissions are sought. Specify any local guidelines (e.g. from local professional associations/learned societies/universities) that exist and whether these involve any ethical stipulations beyond those usual in the UK (provide details of any licenses or permissions required). Also specify whether there are any specific ethical issues raised by the local context in which the research activity is taking place, for example, particular cultural and/or legal sensitivities or vulnerabilities of participants.
13	<i>(this box should expand as you type)</i>

14	Use of documentation not in the public domain: Are any documents NOT publicly available?	NO	☒
		YES	☐
15	If Yes, please provide details here of how you will gain access to specific documentation that is not in the public domain and that this is in accordance with prevailing data protection law of the country in question and England and Wales. <i>(this box should expand as you type)</i>		

SECTION F: Scope of Research Activity

1	Will the research activity include:	YES	NO
2	Use of a questionnaire or similar research instrument?	☐	☒
3	Use of interviews?	☐	☒
4	Use of diaries?	☐	☒
5	Participant observation with their knowledge?	☐	☒
6	Participant observation without their knowledge?	☐	☒
7	Use of video or audio recording?	☐	☒
8	Access to personal or confidential information without the participants' specific consent?	☐	☒
9	Administration of any questions, test stimuli, presentation that may be experienced as physically, mentally or emotionally harmful / offensive?	☐	☒
10	Performance of any acts which may cause embarrassment or affect self-esteem?	☐	☒
11	Investigation of participants involved in illegal activities?	☐	☒
12	Use of procedures that involve deception?	☐	☒
13	Administration of any substance, agent or placebo?	☐	☒
14	Working with live vertebrate animals?	☐	☒
15	Other primary data collection methods, please explain in this box	☐	☒

	For example, 'focus groups'. Please indicate the type of data collection method(s) in this box and tick the accompany box.		
16	Details of any other primary data collection method: (this box should expand as you type)		

If NO to every question, then the research activity is (ethically) low risk and **may** be exempt from **some** of the following sections (please refer to Guidance Notes).

If YES to any question, then no research activity should be undertaken until full ethical approval has been obtained.

SECTION G: Intended Participants

1	Who are the intended participants:	YES	NO
2	Students or staff at the University?	☐	☒
3	Adults (over the age of 18 and competent to give consent)?	☐	☒
4	Vulnerable adults?	☐	☒
5	Children and Young People under the age of 18? (Consent from Parent, Carer or Guardian will be required)	☐	☒
6	Prisoners?	☐	☒
7	Young offenders?	☐	☒
8	Those who could be considered to have a particularly dependent relationship with the investigator or a gatekeeper?	☐	☒
9	People engaged in illegal activities?	☐	☒
10	Others (please identify specifically any group who may be unable to give consent) please indicate here and tick the appropriate box.	☐	☒
11	Other – please indicate here: (this box should expand as you type)		

12	Participant numbers and source Provide an estimate of the expected number of participants. How will you identify participants and how will they be recruited?	
13	How many participants are expected?	NA (this box should expand as you type)
14	Who will the participants be?	NA (this box should expand as you type)
15	How will you identify the participants?	NA (this box should expand as you type)

16	Information for participants:	YES	NO	N/A
17	Will you describe the main research procedures to participants in advance, so that they are informed about what to expect?	☐	☐	☒
18	Will you tell participants that their participation is voluntary?	☐	☐	☒
19	Will you obtain written consent for participation?	☐	☐	☒
20	Will you explain to participants that refusal to participate in the research will not affect their treatment or education (if relevant)?	☐	☐	☒
21	If the research is observational, will you ask participants for their consent to being observed?	☐	☐	☒
22	Will you tell participants that they may withdraw from the research at any time and for any reason?	☐	☐	☒
23	With questionnaires, will you give participants the option of omitting questions they do not want to answer?	☐	☐	☒
24	Will you tell participants that their data will be treated with full confidentiality and that, if published, it will not be identifiable as theirs?	☐	☐	☒
25	Will you debrief participants at the end of their participation, in a way appropriate to the type of research undertaken?	☐	☐	☒
26	If NO to any of above questions, please give an explanation			
27	<i>(this box should expand as you type)</i>			

28	Information for participants:	YES	NO	N/A
29	Will participants be paid?	☐	☐	☒
30	Is specialist electrical or other equipment to be used with participants?	☐	☐	☒
31	Are there any financial or other interests to the investigator or University arising from this study?	☐	☐	☒
32	Will the research activity involve deliberately misleading participants in any way, or the partial or full concealment of the specific study aims?	☐	☐	☒
33	If YES to any question, please provide full details			
34	<i>(this box should expand as you type)</i>			

SECTION H: Anticipated Risks

1	Outline any anticipated risks that may adversely affect any of the participants, the researchers and/or the University, and the steps that will be taken to address them. If you have completed a full risk assessment (for example as required by a laboratory, or external research collaborator) you may append that to this form.						
2	Full risk assessment completed and appended?		<table border="1"> <tr> <td>Yes</td> <td>☐</td> </tr> <tr> <td>No</td> <td>☒</td> </tr> </table>	Yes	☐	No	☒
Yes	☐						
No	☒						
3	Risks to participants For example: emotional distress, financial disclosure, physical harm, transfer of personal data, sensitive organisational information						
4	Risk to Participant: NA <i>(this box should expand as you type)</i>	How will you mitigate the Risk to Participant					

	(this box should expand as you type)		
5	If research activity may include sensitive, embarrassing or upsetting topics (e.g. sexual activity, drug use) or issues likely to disclose information requiring further action (e.g. criminal activity), give details of the procedures to deal with these issues, including any support/advice (e.g. helpline numbers) to be offered to participants. Note that where applicable, consent procedures should make it clear that if something potentially or actually illegal is discovered in the course of a project, it may need to be disclosed to the proper authorities		
	(this box should expand as you type)		
6	Risks to investigator For example: personal safety, physical harm, emotional distress, risk of accusation of harm/impropriety, conflict of interest		
	Risk to Investigator: Potential risks for investigators include data privacy and security breaches, emotional distress, accusations of harm or impropriety, conflicts of interest, intellectual property disputes, and exposure to cybersecurity threats. Mitigation strategies include ethical guidelines, data protection, legal advice, transparent communication, and addressing conflicts of interest. (this box should expand as you type)	How will you mitigate the Risk to Investigator: To mitigate risks to investigators, we will establish clear ethical guidelines, prioritize data privacy and security, seek legal advice when needed, and maintain transparent communication with stakeholders. Furthermore, we will uphold professional integrity, promptly acknowledge and address conflicts of interest, and ensure a secure research environment. (this box should expand as you type)	
7	University/institutional risks For example: adverse publicity, financial loss, data protection		
	Risk to University: Possible risks to the university if this research is conducted include adverse publicity, financial loss, data protection breaches, legal liabilities, damage to institutional reputation, and potential conflicts of interest. These risks can arise from mishandled sensitive data, unethical research practices, or perceived impropriety in the research process. (this box should expand as you type)	How will you mitigate the Risk to University: To mitigate risks to the university, we will adhere to stringent ethical guidelines, prioritize data privacy and security, and ensure compliance with relevant laws and regulations. Regular progress reporting and transparent communication will be maintained with university stakeholders. Additionally, we will proactively address potential conflicts of interest and strive to uphold the institution's reputation through rigorous research practices. (this box should expand as you type)	

8	Disclosure and Barring Service			
9	If the research activity involves children or vulnerable adults, a Disclosure and Barring Service (DBS) certificate must be obtained before any contact with such participants.	YES	NO	N/A
10	Does your research require you to hold a current DBS Certificate?	☐	☐	☒

SECTION I: Feedback, Consent and Confidentiality

1	Feedback What de-briefing and feedback will be provided to participants, how will this be done and when? NA (this box should expand as you type)
2	Informed consent

	Describe the arrangements to inform potential participants, before providing consent, of what is involved in participating. Describe the arrangements for participants to provide full consent before data collection begins. If gaining consent in this way is inappropriate, explain how consent will be obtained and recorded in accordance with prevailing data protection legislation.
	NA <i>(this box should expand as you type)</i>
3	Confidentiality / Anonymity Set out how anonymity of participants and confidentiality will be ensured in any outputs. If anonymity is not being offered, explain why this is the case.
	NA <i>(this box should expand as you type)</i>

SECTION J: Data Protection and Storage

In completing this section refer to the University's Research Data Management Policy and the extensive resources on the University's Research Data Management web pages (<http://uwtsd.ac.uk/library/research-data-management/>).

1	Does the research activity involve personal data (as defined by the General Data Protection Regulation 2016 "GDPR" and the Data Protection Act 2018 "DPA")?	YES	NO
	"Personal data" means any information relating to an identified or identifiable natural person ('data subject'). An identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person.	☐	☒
2	If YES, provide a description of the data and explain why this data needs to be collected:		
	<i>(this box should expand as you type)</i>		
3	Does it involve special category data (as defined by the GDPR)?	YES	NO
	"Special category data" means sensitive personal data consisting of information as to the data subjects' – (a) racial or ethnic origin, (b) political opinions, (c) religious beliefs or other beliefs of a similar nature, (d) membership of a trade union (within the meaning of the Trade Union and Labour Relations (Consolidation) Act 1992), (e) physical or mental health or condition, (f) sexual life, (g) genetics, (h) biometric data (as used for ID purposes),	☐	☒
4	If YES, provide a description of the special category data and explain why this data needs to be collected:		
	<i>(this box should expand as you type)</i>		
5	Will the research activity involve storing personal data and/or special category data on one of the following:	YES	NO

6	Manual files (i.e. in paper form)?	☐	☒
7	University computers?	☐	☒
8	Private company computers?	☐	☒
9	Home or other personal computers?	☐	☒
10	Laptop computers/ CDs/ Portable disk-drives/ memory sticks?	☐	☒
11	"Cloud" storage or websites?	☐	☒
12	Other – specify:	☐	☒
13	For all stored data, explain the measures in place to ensure the security of the data collected, data confidentiality, including details of password protection, encryption, anonymisation and pseudonymisation:		
	<i>(this box should expand as you type)</i>		
14	All Data Storage		
15	Will the research activity involve any of the following activities:	YES	NO
16	Electronic transfer of data in any form?	☒	☐
17	Sharing of data with others at the University?	☒	☐
18	Sharing of data with other organisations?	☐	☒
19	Export of data outside the European Union or importing of data from outside the UK?	☒	☐
20	Use of personal addresses, postcodes, faxes, emails or telephone numbers?	☐	☒
21	Publication of data that might allow identification of individuals?	☐	☒
22	Use of data management system?	☐	☒
23	Data archiving?	☐	☒
24	If YES to any question, please provide full details, explaining how this will be conducted in accordance with the GDPR and DPA (and/or any international equivalent):		
	All data will be encrypted and kept in password protected cloud storage on the University Office 365 system which will not be shared. Any USB sticks used to store or transfer data will be password protected. All data transfers will be encrypted and password protected. All participants will be given a unique identifier to ensure confidentiality and this list will be kept securely in the password protected folder. The data will be stored until the completion of the project and then deleted. <i>(this box should expand as you type)</i>		
25	List all who will have access to the data generated by the research activity:		
	The principal researcher and the supervisor. <i>(this box should expand as you type)</i>		
26	List who will have control of, and act as custodian(s) for, data generated by the research activity:		
	The principal researcher. <i>(this box should expand as you type)</i>		

27	Give details of data storage arrangements, including security measures in place to protect the data, where data will be stored, how long for, and in what form. Will data be archived – if so how and if not why not.		
	All data will be encrypted and kept in password protected cloud storage on the University Office 365 system which will not be shared. Any USB sticks used to store or transfer data will be password protected, and will be reformatted at the end of the project in order to destroy the data. The data will be stored until the completion of the project and then deleted. <i>(this box should expand as you type)</i>		
28	Please indicate if your data will be stored in the UWTSD Research Data Repository (see https://researchdata.uwtsd.ac.uk/). If so please explain. <i>(Most relevant to academic staff)</i>		
	No <i>(this box should expand as you type)</i>		
29	Confirm that you have read the UWTSD guidance on data management (see https://www.uwtsd.ac.uk/library/research-data-management/)	YES	☒
		NO	☐
30	Confirm that you are aware that you need to keep all data until after your research has completed or the end of your funding	YES	☒
		NO	☐

SECTION K: Declaration

31	The information which I have provided is correct and complete to the best of my knowledge. I have attempted to identify any risks and issues related to the research activity and acknowledge my obligations and the rights of the participants.		
	In submitting this application I hereby confirm that I undertake to ensure that the above named research activity will meet the University's Research Ethics and Integrity Code of Practice which is published on the website: https://www.uwtsd.ac.uk/research/research-ethics/		
	Signature of applicant:	Zhiqi Lu	Date: Sep 15, 2023

For STUDENT Submissions:

32	Director of Studies/Supervisor:		Date:
33	Signature:		

For STAFF Submissions:

34	Academic Director/ Assistant Dean:		Date:
35	Signature:		

Checklist: Please complete the checklist below to ensure that you have completed the form according to the guidelines and attached any required documentation:

☒	I have read the guidance notes supplied before completing the form.
☒	I have completed ALL RELEVANT sections of the form in full.
☒	I confirm that the research activity has received approval in principle
☒	I have attached a copy of final/interim approval from external organisation (where appropriate)
☒	I have attached a full risk assessment (and have NOT completed Section H of this form) (where appropriate) ONLY TICK IF YOU HAVE ATTACHED A FULL RISK ASSESSMENT
☒	I understand that it is my responsibility to ensure that the above named research activity will meet the University's Research Ethics and Integrity Code of Practice.
☒	I understand that before commencing data collection all documents aimed at respondents (including information sheets, consent forms, questionnaires, interview schedules etc.) must be confirmed by the DoS/Supervisor, module tutor or Academic Director.
☒	I have deleted the guidance notes before submitting the PG2 for consideration

RESEARCH STUDENTS AND STAFF ONLY

All communications relating to this application during its processing must be in writing and emailed to pgresearch@uwtsd.ac.uk, with the title 'Ethical Approval' followed by your name. You will be informed of the outcome of your claim by email; therefore it is important that you check your University and personal email accounts regularly.

STUDENTS ON UNDERGRADUATE OR TAUGHT MASTERS PROGRAMMES should submit this form (and receive the outcome) via systems explained to you by the supervisor/module leader.