Exploring the use of Explainable AI for improving intrusion detection systems

MSc Research Project
MSc Cybersecurity

Ravi Ranjan Singh
Student ID: x22203052

School of Computing
National College of Ireland

Supervisor: Michael Prior

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	**Ravi Ranjan Singh**		
Student ID:	**X22203052**		
Programme:	**MSc Cybersecurity**	**Year:**	**Jan 2024-2025**
Module:	**MSc Research Project**		
Supervisor:	**Michael Prior**		
Submission Due Date:	**12/12/2024**		
Project Title:	**Exploring the use of Explainable AI for improving intrusion detection systems**		
Word Count:	**5636 Page Count 25**		

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature: **Ravi Ranjan Singh**

Date: **11/12/2024**

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. **Please attach a completed copy of this sheet to each project (including multiple copies).**
2. **Projects should be submitted to your Programme Coordinator.**
3. **You must ensure that you retain a HARD COPY of ALL projects, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.**
4. **You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. Late submissions will incur penalties.**
5. **All projects must be submitted and passed in order to successfully complete the year. Any project/assignment not submitted will be marked as a fail.**

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

AI Acknowledgement Supplement

MSc Research Project
Exploring the use of Explainable AI for improving intrusion detection systems

Your Name/Student Number	Course	Date
Ravi Ranjan Singh	MSc Cybersecurity	11/12/2024

This section is a supplement to the main assignment, to be used if AI was used in any capacity in the creation of your assignment; if you have queries about how to do this, please contact your lecturer. For an example of how to fill these sections out, please click here.

AI Acknowledgment

This section acknowledges the AI tools that were utilized in the process of completing this assignment.

Tool Name	Brief Description	Link to tool
NA	NA	NA

Description of AI Usage

This section provides a more detailed description of how the AI tools were used in the assignment. It includes information about the prompts given to the AI tool, the responses received, and how these responses were utilized or modified in the assignment. One table should be used for each tool used.

[Insert Tool Name]	
NA	
NA	NA

Evidence of AI Usage

This section includes evidence of significant prompts and responses used or generated through the AI tool. It should provide a clear understanding of the extent to which the AI tool was used in the assignment. Evidence may be attached via screenshots or text.

Additional Evidence:

[Place evidence here]

Additional Evidence:

[Place evidence here]

Exploring the use of Explainable AI for improving intrusion detection systems

Ravi Ranjan Singh
x22203052

Abstract

The aim and objectives of this research therefore lies in the proposition of a solution to the global challenge of enhancing IDSs through the employment of deep ML classifiers as Random Forest, Support Vector Machine, or Multi-Layer Perceptron. SVM was found to have an accuracy of 82% with comparable precision and recall making it superior to all models in this study. On the other hand, MLP detected 95% of the time for malicious traffic and achieved the highest accuracy of 84% in differentiating benign traffic. In order to ensure that outputs are clearly and transparently interpretable, this study highlighted the features responsible for critical decisions about model selections with SHAP and LIME. The findings highlight that intrusion detection system facilitated through the integration of artificial intelligence significantly enhances cybersecurity through the delivery of trusted and explainable intrusion detection services. Areas that may be explored in future research so as to increase efficacy include real time monitoring and ideal model configurations.

1 Introduction

1.1 Background and Motivation

Intrusion detection systems (IDS) are one of the most promising and rapidly developing means of protecting computer networks against intrusions in the past several years. False negatives or false positives are often a consequence of a core characteristic of classical IDSs that work based on a priori known patterns and rules that define malicious behaviour. These systems are even more flexible and precise now that data patterns are worked on by machine learning (Mane and Rao, 2021). Some of these models have impacts that are hard to interpret especially in the modern complex models such as deep learning where the program's decision-making process is difficult to explain, this leads to detriment in the adoption of the. The answer that comes out is explainable AI (XAI), which gives people the means to understand how it concludes. In this way, the people may comprehend why a system deemed some actions as invasions (Mahbooba *et al.*, 2021). By being applied to IDS, XAI has the potential to increase the detection rates by eliminating the murky middle and enhancing the response strategies by making the IDS's conclusions credible and comprehensible to the security teams. XAI is explored in this study about its role in enhancing IDS performance.

1.2 Research Objectives

- To develop and implement Explainable AI (XAI) techniques within intrusion detection systems (IDS) to enhance the detection and interpretation of cyber threats.
- To analyze and document the current limitations of traditional intrusion detection systems in terms of explainability and transparency.
- To investigate the impact of implementing Explainable AI in IDS on the identification and reduction of false positives and false negatives in cybersecurity threat detection.
- To assess the effectiveness of various XAI methods, such as LIME and SHAP, in providing clear and actionable insights for IDS outputs.
- To conduct a comparative analysis of LIME and SHAP methods in generating explainable AI-powered IDS, examining their influence on the decision-making process of security analysts.

1.3 Research Questions

- How can Explainable AI (XAI) techniques, such as SHAP and feature importance analysis, be effectively integrated into Intrusion Detection Systems (IDS) to enhance cyber threat detection, while ensuring transparency and interpretability for security analysts in real-time decision-making?
- What are the comparative benefits and limitations of using LIME versus SHAP for generating explanations in AI-powered Intrusion Detection Systems, and how do these methods influence the decision-making process of security analysts?

1.4 Assumption and limitation:

In the following discussion, we make several assumptions that are indeed extremely influential in our study. First, it assumes that attack patterns and network traffic records that are used as the basis for training the Intrusion Detection Systems (IDS) are exhaustive, truthful, and of premium quality. Moreover, it assumes that broad practices of XAI techniques such as SHAP or LIME are applicable to identify the rationale behind the behavior and decisions of the complex artificial neural networks applied in IDSs.

Despite these benefits, this study does however have some drawbacks. Since the assaults that were studied might have been scoped to these types the results may not transfer to other types of intrusion such as phishing or denial of service. Moreover, most XAI methods are specific to particular models, meaning, the explanations derived are not transferable to other ML platforms.

1.5 Problem Statement

IDS should be strong and transparent as the cyber-threat is rising significantly day by day. Although making specific detection more accurate, classical machine learning techniques are not interpretable while providing security analysts with no actionable insights. XAI's purpose enhances IDS decision trust through the mechanisms of explaining what has been done. For instance, only a limited number of cybersecurity researchers compare SHAP and LIME XAI techniques. To address this gap, this study analyses their effectiveness in improving the understanding of IDS and decreasing false positive/negative rates. This work enhances IDS with XAI and machine learning models for timely and meaningful cybersecurity information.

1.6 Structure of the Document:

The second part, "Related work," presents an array of previous academic research that is relevant to the subject of this study. Section 3 will be marked by the Research Method and will state the process and the specific plan. In the next Section 4, the Design Specification is provided following the framework of architecture, development process, and technical requirements to build the IDS. This we will do in Section 5 of this paper which briefly describes the Implementation that encompasses how the models run and their accuracies. The relation and difference between machine learning and deep learning will be reviewed in section 6 Evaluation of models. Last of all, in the seventh and final section of the paper, suggestions for future research will be presented.

2 Related Work

1.1 Introduction to Intrusion Detection Systems

According to Ahmad *et al.*, (2021), the size of networks and the data associated with them are increasing rapidly because of the progress achieved in the development of the Internet and other forms of communication. Protection of the networks raises the concern that the creators of malicious attacks are increasingly using new forms of invasions, which makes it almost impossible to diagnose them with some level of precision. Such a tool is an intrusion detection system or IDS, which scans the flow of traffic for signs of intrusion besides doing other things to make sure that the network is safe, private, and available at all times. While a lot of work has been done on IDS there is much more still to be done before it can recognize new intrusions with reasonable accuracy and a low number of false positives. To attempt and offer a way to better scope intrusions across the network, IDS systems utilizing machine learning (ML) and deep learning (DL) have recently been employed. IDS is introduced and explained in the article before moving on to the construction of a taxonomy based on the most popular MDL methods in creating NIDSS systems.

Whereas Ozkan-Okay *et al.*, (2021), states that computer network intrusion detection is still quite problematic. Cyber attackers camouflage themselves behind injecting contents within the packets which breach the intrusion detection system (IDS). Moreover, new computer networks are being equipped with numerous extra apparatus on a literal day to day basis. These new kinks also hamper the security of the computer networks. In order to effectively manage the computer network traffic and to offer security ahead of time, both the components, methods and technologies, threats and tools of the IDS have to be examined.

2.2 Machine Learning Approaches to Network Intrusion Detection

As per Kilincer *et al.*, (2021), the internet is being used by people, it procreates and with increased users comes increased security threats. Since these are security vulnerabilities in the systems, they permit the control of the way the systems function and breach the privacy of data. To trace and record intrusion attempts, concepts called Intrusion Detection Systems (IDS) were developed. This research reviews in detail the existing work that utilizes the CSE-CIC IDS-2018, UNSW-NB15, ISCX-2012, NSL-KDD, and CIDDS-001 datasets. These datasets are widely used while designing intrusion detection systems. In addition, the presented datasets went through max–min normalization and classification using traditional machine learning algorithms including SVM, KNN, and DT. As a result, some of the trials reflected in the literature have been able to record better results.

Whereas Rabbani *et al.*, (2021), states that NADSs can monitor for and halt dangerous activity, they are an inherent component of any security network. Therefore, this article offers a broad survey of various aspects of NIDSSs developed based on an anomaly detection technique. Also provided are the major characteristics of intrusion detection systems and the most current malicious activities in network systems. In this survey, major stages of NADSs such as preprocessing, feature extraction, and identification and detection of injurious activity are described. In addition, the researchers have provided some information about the current benchmark datasets for training and testing of the machine learning techniques and the researchers have explained the recent machine learning techniques like supervised, unsupervised, new deep learning, ensemble, and detection and recognition phase in detail.

Although Alzahrani and Alenazi, (2021), is of the view that Software-defined Networking (SDN), bring optimism to the future of the Internet. With SDN a more managed, centralized, and controlled network has been made easier and more open. These issues are catastrophic to enterprises, economies, and organizations in general. Over the last ten years, several applications for SDNs have aimed at extending advanced machine learning techniques into network intrusion detection systems (NIDS). This study demonstrates how SDN controller NIDS performs traffic analysis using machine learning techniques for malicious activity detection. Attack detection is elaborated with Decision Tree, Random Forest, and XGBoost. The NSL-KDD dataset is utilized to train and evaluate the proposed methods and compare various advanced NIDS techniques.

2.2.1 Deep Learning Approaches to Network Intrusion Detection

As per Gamage and Samarabandu, (2020), the deep learning methodologies applied in the domain of intrusion detection are the areas under research in cybersecurity. Numerous excellent surveys cover the growing body of the literature regarding this problem, however, the reports are missing a critical evaluation of the performance of several deep learning models under a similar context, especially about the modern datasets for intrusion detection. Four major models including a feed-forward neural network, autoencoder, deep belief network, and long short-term memory network are trained for the task of intrusion detection on two older datasets (KDD 99, NSL-KDD) and two contemporary ones. The authors observed that deep-feedforward neural networks have very high performance scores when it comes to accuracy, F1 score, training, and inference time, across the four datasets. The semi-supervised approaches such as autoencoders and deep belief networks did not outperform traditional supervised feed-forward neural networks.

According to Su *et al.*, (2020), states that unknown attacks from network traffic can be detected by network intrusion detection and is efficient as well. KNN, SVM, etc, are some of the algorithms employed by the current network anomaly detection methods. Although this type of method can be associated with various impressive features, they do produce some effective results but are inaccurate, as they depend heavily on manual crowd traffic features engineering, which is too backward for the big data era. Low detection and feature engineering problems associated with classifying traffic anomalies detection technique BAT enhances performance in intrusion detection. The 2nd and 3rd components of the BAT model comprise the BLSTM model and attention mechanism respectively. The BAT model in M-C form explains the attached author's achievements as they say data samples are processed using scripted convolutional networks and the network traffic is classified using Softmax.

Similarly, Ashiku and Dagli, (2021), is of the view that the evolution or improvement of systems has invariably depended on the integration and universal use of computing systems and this has been of great benefit to mankind. This paper proposes the construction of new Deep Learning-based Intrusion Detection Systems (IDS) to detect and classify the attacks on the network. The application of deep neural networks (DNNs) to develop such adaptive IDS, which could perform the dual function of assessing biological behaviour patterns with previously known and new intrusions in the system, is what this paper examines. The study used the UNSW-NB15 dataset that demonstrated how the model behaved during a simulation of external aggressive attack activities to demonstrate how the model replicated the real nature of contemporary networks.

Whereas Imrana *et al.*, (2021), states that most of the service providers are anxious over the increase of computer networks coupled with the internet threats. It has led to the research and implementation of intrusion detection systems (IDSs) to prevent and control intrusions into the network. For many years, intrusion detection systems have been able to detect network attacks and anomalies. Intrusion detection systems have been proposed by many researchers in different countries to deal with network intruders. Many of the proposed IDSs do have a good performance in terms of false positives.

According to Aldweesh *et al.*, (2020), higher intrusion detection systems (IDSs) is necessary because of the critical security threats brought about the rapid increase of data being transmitted through various devices and channels. More advanced models of learning, like Deep Learning, structure a learning process around the connection of, and computation in, a large number of units (neurons) in a network. There's a plethora of other applications where deep learning does the trick especially when data is in abundance. It's no surprise then that deep learning for intrusion detection has been attracting a lot of researchers in recent times. Some of the most relevant prior polls in the field of cybersecurity conducting deep learning are analyzed and compared in this research work.

Table 1: Comparative analysis

Study (Author, Year)	Purpose	Key Features	Strengths	Weaknesses	Proposed Algorithm/Approach
Ahmad et al. (2021)	Overview of IDS, challenges in intrusion detection with emerging threats	Discusses advancements in IDS, use of machine learning, and deep learning	Highlights the evolving challenges with intrusion detection due to network growth	Limited coverage on specific ML/DL techniques and new attack scenarios	Utilization of ML/DL for NIDSS
Kilincer et al. (2021)	Review of AI-based IDS methods using various datasets	Comparison of datasets (CSE-CIC, UNSW-NB15, ISCX-2012, NSL-KDD) and classifiers like SVM, KNN, and DT	Effectiveness of AI techniques for intrusion detection	Limited real-time adaptability in existing systems	AI techniques using ML models for improved detection
Rabbani et al. (2021)	Survey of NADS using anomaly detection techniques	Analysis of NADS stages (preprocessing, feature extraction, detection), exploration of machine learning techniques	Thorough examination of feature extraction and detection stages	Lack of evaluation of specific novel datasets	Supervised and unsupervised ML algorithms for NADS
Alzahrani & Alenazi (2021)	Evaluation of SDN-based NIDS using machine learning methods	Use of SDN and advanced AI techniques (DT, RF, XGBoost) for detection	Highlights advantages of SDN for centralization and control in networks	Focuses on SDN; less emphasis on general NIDS solutions	Machine learning for NIDS with Decision Tree, Random Forest, and XGBoost
Gamage & Samarabandu (2020)	Deep learning methods for	Comparison of deep learning models (FFNN,	Detailed performance metrics for DL models	Limited focus on contemporary datasets beyond KDD	Deep learning models with a focus on FFNN, Autoencoder,

	intrusion detection	Autoencoder, DBN, LSTM) across modern and older datasets		99, NSL-KDD	DBN, and LSTM for NIDSS
Su et al. (2020)	Enhancement of anomaly detection through BAT (BLSTM and attention mechanisms)	Introduction of BAT model with BLSTM and attention mechanism for traffic classification	Captures hierarchical importance of features, effective anomaly detection	Dependent on proper training data; computationally intensive	BAT model with BLSTM and attention mechanisms for network traffic analysis
Ashiku & Dagli (2021)	Creation of a Deep Learning-based adaptive IDS	Use of DNN for assessing biological behavior patterns and classifying known and unknown intrusions	Demonstrates adaptability to real-world aggressive attacks in simulations	Model behavior dependent on dataset-specific attributes (UNSW-NB15)	DNN for real-time assessment and adaptive IDS for dynamic threat detection
Ozkan-Okay et al., (2021)	To examine the history, development, components, methods, and technologies of intrusion detection systems (IDS).	Explores ISS robust capabilities, IDS components and technologies, datasets, popular tools, types of attacks, and defense mechanisms.	Comprehensive discussion on IDS capabilities and threats, modern approaches, and tools.	Generalized discussion, lacks focus on specific model implementation or dataset evaluation.	None explicitly mentioned; focuses on the holistic analysis of IDS technologies.
Imrana et al., (2021)	To analyze the challenges of intrusion detection systems	Investigates false positives, anomaly detection, and the difficulty in	Effective discussion of false positives and anomaly detection.	Limited focus on U2R and R2L attacks; does not propose effective	None explicitly mentioned; highlights the need for better IDS models for U2R and R2L assaults.

	(IDS) in detecting U2R and R2L assaults.	detecting specific attacks (U2R and R2L).		solutions to improve detection accuracy.	
Aldweesh et al., (2020)	To analyze and compare the use of deep learning in intrusion detection systems.	Reviews deep learning applications for intrusion detection, focusing on its benefits with abundant data.	Highlights the effectiveness of deep learning models and their growing research interest in cybersecurity.	Lacks detailed comparison metrics or experimental validation for the discussed deep learning models.	Suggests deep learning models as effective tools for intrusion detection.

3 Research Methodology

3.1 Methodology

The methodological framework of CRISP-DM is one of the most widely used in the retrieval of knowledge from data. It was specifically designed by a set of industry experts towards the end of the 1990s who sought to assist many such professionals navigate the terrain of data mining. The CRISP-DM consists of 6 processes. Business Understanding commences with comprehension of the organization’s objectives including how the data mining process will be contributory (Schröer *et al.*, 2021). Second refers to Data Understanding where data is gathered, examined, or interpreted and some useful definitions are made to help in addressing the quality issues. In this stage, the data preparation process, all unstructured, raw data is refined, structured, and formatted for modeling purposes (Saltz, 2021). Fourth in the sequence, Modeling is placing all or some of the prepared data into models to describe and predict the target variable; this is just a verbal rendition of a data modeling technique. In the next phase, the Evaluation phase, the performance of the model is determined in terms of the established objectives and how well the model was built or constructed.

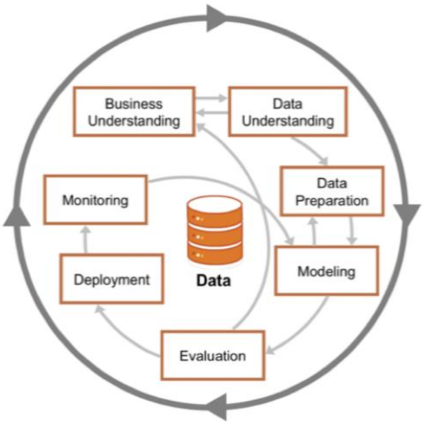


Figure 1: CRISP-DM Framework

Source: (Saltz, 2021)

Business Understanding:

In this research, business understanding suggests identifying the business problem of improving the IDS using Explainable AI. The study aims to identify possible network attacks and to offer clear and reasonable explanations of the choices made during the detection process.

Data Understanding:

Data understanding comprehends searching for relevant datasets appropriate for IDS model training purposes. In this current study, an NSL-KDD99 dataset will serve as a means of scheming over the network traffic to seek out anomalies (kaggle.com, 2023). IT diagnosis will focus on the structure of the dataset, behaviors that are normal and anomaly or abnormal, and patterns. The type of protocol, source/destination (SD) IP, packet size, and type of traffic are analyzed at this stage.

Data Preparation:

Understanding the data is followed in order by preparing the data for modeling. In this phase, data cleaning is conducted, which eliminates unnecessary features and handles missing values. Few categorical variables such as protocol type are normalized or encoded for the model to be prepared.

Modeling:

Random Forest, SVM, and neural networks like MLP Classifier are used to create prediction models for IDS. The emphasis is thus on the creation of interpretable and efficient models. For these models explanations SHAP and LIME will be used to explain model behavior and how the system performs intrusion detection respectively.

Evaluation:

Compelling and informative systems adhere to classification model accuracy, precision, recall, and F-1 score in a zero system after the models have been developed. The blind spot of the proposed XAI tool model is the interpretability of the output information obtained after the model is applied. This phase determines the quality of the model in terms of reducing the false positive rate while generating human-understandable explanations for the model's predictions.

Deployment:

In the deployment phase, the web application is developed for network intrusion detection in which when network data is uploaded it shows the result whether it consists of an attack network or a normal network and also it shows SHAP value. The models are being trained first with the NSL-KDD data then based on the best result generated from the three models, one model is selected and saved and then implemented in the web application.

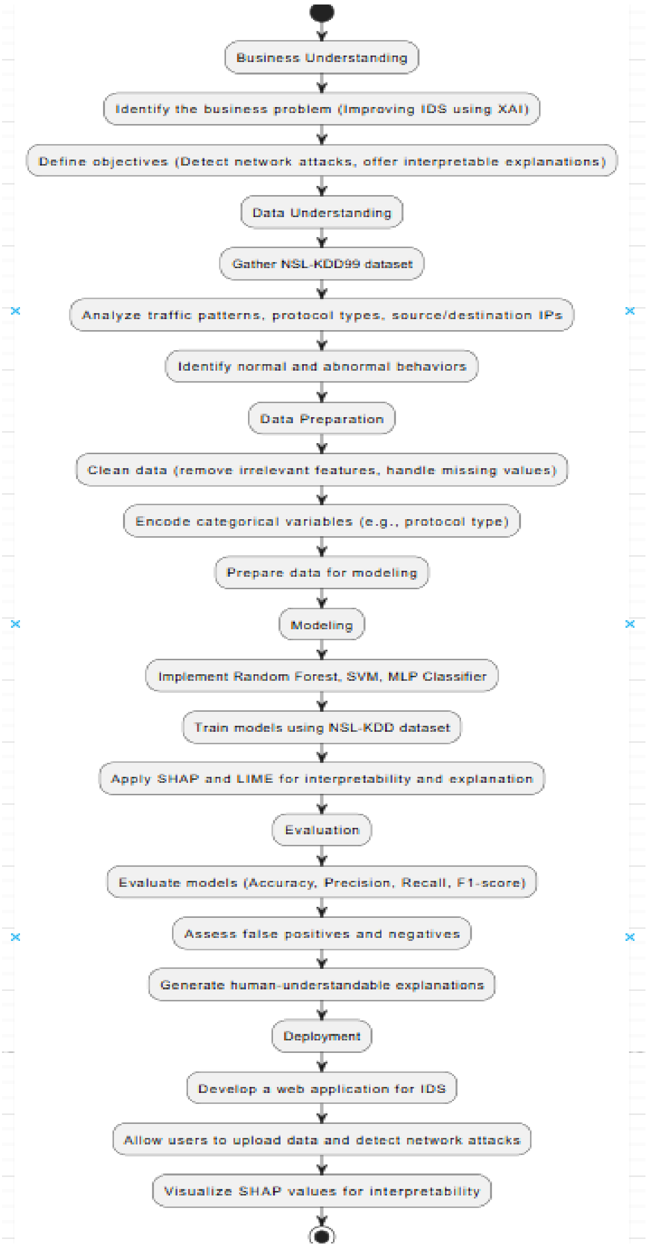

Business Understanding
Identify the business problem (Improving IDS using XAI)
Define objectives (Detect network attacks, offer interpretable explanations)
Data Understanding
Gather NSL-KDD99 dataset
Analyze traffic patterns, protocol types, source/destination IPs
Identify normal and abnormal behaviors
Data Preparation
Clean data (remove irrelevant features, handle missing values)
Encode categorical variables (e.g., protocol type)
Prepare data for modeling
Modeling
Implement Random Forest, SVM, MLP Classifier
Train models using NSL-KDD dataset
Apply SHAP and LIME for interpretability and explanation
Evaluation
Evaluate models (Accuracy, Precision, Recall, F1-score)
Assess false positives and negatives
Generate human-understandable explanations
Deployment
Develop a web application for IDS
Allow users to upload data and detect network attacks
Visualize SHAP values for interpretability

Figure 2: Flow Chart

3.2 Libraries Imported

Several Python packages are available in the development and testing of machine learning models for the detection of intrusions. The data frame format used in the machine is dealt with by Pandas, which makes it easy to import, clean, and manipulate the data. The most well-known machine learning framework. sklearn contains methods for constructing classification models such as RandomForestClassifier and SGDClassifier. In this case, the Standard scale normally is used to sin and to understand the performance of the regression analysis metric. F1, accuracy, precision, and recall score were used to determine the statement of each of the cases and its performance.

The basic matrix and array operations that are useful for math modules for learning algorithms in machine learning are organized by the numpy package. Plots and charts performed by Matplotlib help depict both the dataset and the efficiency of the model.

3.3 Feature Extraction

Feature extraction is a fundamental process in machine learning where raw information is efficiently captured into more efficient and simpler features that are used when training a model. For instance, in network intrusion detection systems (NIDS), feature extraction again helps to capture important aspects of particular network traffic, for example, the amount of data packets, the types of protocols used, or the connection time to seek out abnormal or malicious activities. This method reduces the size of the dataset, while at the same time keeping its more informative parts to enhance the accuracy and performance of the model. Proper feature extraction mitigates redundancy and irrelevant information enabling faster model training. To describe the variations of data patterns in a dataset, principal component analysis (PCA) statistical techniques and other techniques that are relevant to the domain are utilized to select and extract relevant features enhancing the models for machine learning.

3.4 Data Splitting (Training and Testing the Model)

Data splitting involves partitioning the data into train and test sets. The portion of the data known as the training set teaches the machine learning model the patterns and relationships within the data. To avoid overfitting and in pursuit of generalization, the testing set assesses the model on data that the model has not seen before. The data set is allocated in the ratio of 80:20 where 80% is for training while 20% is for testing. Their validity in terms of the model accuracy, precision, recall, and other performance measures is justified after this paradigm.

3.5 Dataset Description

The present research employs the NSL-KDD as the KDD 99 dataset for intrusion detection systems in a more precise manner (kaggle.com, 2023). With the enhancements made in the NSL-KDD, excessive records and data imbalance present in the normal dataset are removed creating a dataset that is fair and whole in regards to evaluation of the machine learning algorithms. It contains a range of network traffic both benign and malicious including DoS, Probe, U2R, and R2L attacks. The dataset is useful for training and evaluation of intrusion detection systems because it contains normal and intrusive traffic, as well as useful attributes like protocol type, service, and traffic statistics.

3.6 Justification for Model Choices

For improved and explainable intrusion detection, the authors employed Random Forest, SVM, and MLP Classifier. For this reason, Random Forest is a good anomaly detector based on accuracy and data skewness. Since separation of normal and malicious traffic involves linear as well as non-linear separation, accuracy of SVM is excellent. MLP Classifier, a neural network model, is capable of perceiving complex data trend, and the method can remember anomalies effectively. These models provide a moderate level of interpretability, scalability, and model performance while two models are more focused on specifically explaining results for security analysts, these are SHAP and LIME.

4 Design Specification

Following the NSL-KDD data set begins with the extraction of features that need to be retrieved to tell what is normal network traffic and what entails an intrusion in the network. Thereafter the data sets with the missing values and noise are cleaned and preprocessed to make the data set ready for further investigation. To analyze features and their distribution and relationships, Exploratory Data Analysis (EDA) is performed. After that EDA, the dataset is partitioned for model development into training and testing datasets. Then, train the model with data using Random Forest, SVM and, Neural Networks. Training of the network intrusion detection models is done on the training dataset. The accuracy and precision of the models are evaluated on the testing dataset. Finally, after the models have been validated, they have been embedded in a web application developed in PyCharm to detect and classify network activities as either attacks or normal behaviour.

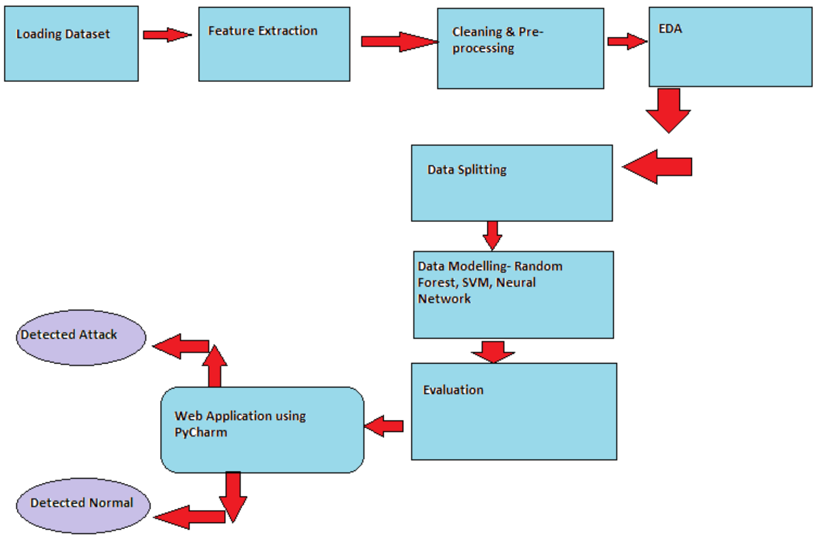

Loading Dataset
Feature Extraction
Cleaning & Pre-processing
EDA
Data Splitting
Data Modelling- Random Forest, SVM, Neural Network
Evaluation
Web Application using PyCharm
Detected Attack
Detected Normal

Figure 3: Workflow

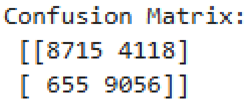
5 Implementation

5.1 Random Forest Classifier

Table 1: Model Evaluation of Random Forest Classifier

Class	Precision	Recall	F1-Score	Support
0	0.93	0.68	0.79	12833
1	0.69	0.93	0.79	9711
Accuracy			**0.79**	22544
Macro Avg	0.81	0.81	0.79	22544
Weighted Avg	0.83	0.79	0.79	22544

A classification report assesses the intrusion detection capability of the Random Forest Classifier. Model accuracy is 79%, meaning they are in most cases correct. The class 0 traffic (which is most probably regular) has also been given a 93% precision, which is rather trustworthy. The recall figures stand at 68% which means that some of the benign traffic is erroneously flagged as harmful. It is also observed that the model has a 93% recognition of occurrences of class 1 with an anomaly or abnormal traffic class yet the recognition is said to be 69% which implies that there are some positive errors. The model's balanced F1 score of 0.79 for the two classes indicates that the model has the potential to achieve a balanced accuracy and recall. This study has been able to determine that Explainable AI (XAI) methodologies can assist in correcting false positives and negatives, thereby enhancing the interpretability and confidence in the intrusion detection system.



```
Confusion Matrix:
 [[8715 4118]
 [ 655 9056]]
```

Figure 4: Confusion Matrix of Random Forest

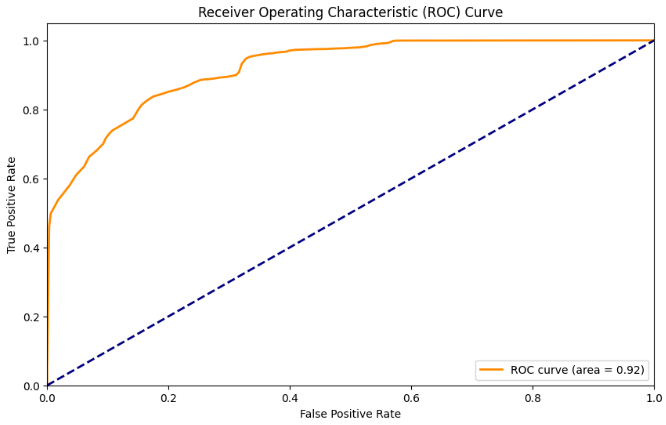


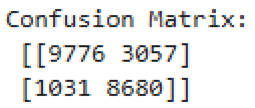
Figure 5: ROC Curve for Random Forest

5.2 Support Vector Machine

Table 2: Model Evaluation of SVM

Class	Precision	Recall	F1-Score	Support
0	0.9	0.76	0.83	12833
1	0.74	0.89	0.81	9711
Accuracy			**0.82**	22544
Macro Avg	0.82	0.83	0.82	22544
Weighted Avg	0.83	0.82	0.82	22544

SVM classifier shows accuracy up to 82%, greater than that of Random Forest model. The overall results show normal traffic (class 0) has an accuracy rate of 90% and recall rate of 76%, which indicates a good reliability factor but still room for improvement in finding every case of normal traffic. The class 1 (anomaly) recall of 89% gives good assurance of detection, however, precision level of 74% implies that there will be some false positives. SVM is robust as evidenced by the F1-scores of both classes which are almost equal. 0.83 and 0.81 that are satisfactory. This study highlights the way SVM can enhance accuracy in intrusion detection.



```
Confusion Matrix:
 [[9776 3057]
 [1031 8680]]
```

Figure 6: Confusion Matrix of SVM

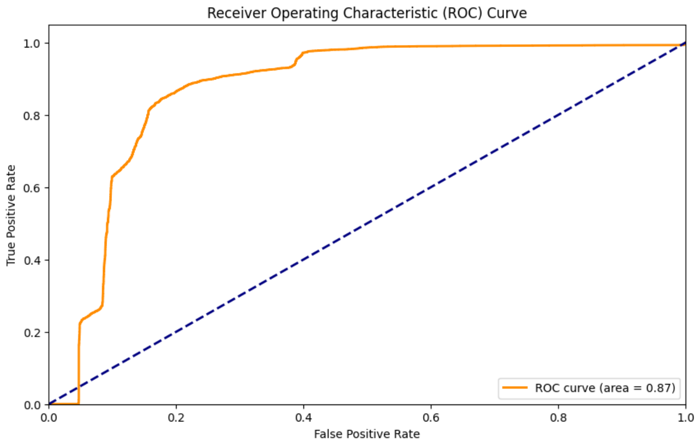


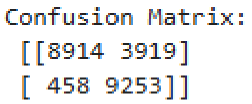
Figure 7: ROC Curve for SVM

5.3 Neural Network: MLP Classifier

Table 3: Model Evaluation of MLP Classifier

Class	Precision	Recall	F1-Score	Support
0	0.95	0.69	0.8	12833
1	0.7	0.95	0.81	9711
Accuracy			**0.81**	22544
Macro Avg	0.83	0.82	0.81	22544
Weighted Avg	0.84	0.81	0.81	22544

The accuracy of the MLP classifier is 81%, providing confidence to its users. For class 0 (normal traffic), the model's precision is 95% but its recall sits at 69% which suggests that some of the benign events are being classified as malicious. In class 1, (anomaly), 95% recall and 70% accuracy are achieved which signifies good detection but also a lot of false alarms. The balanced F1-scores of 0.80 and 0.81 achieved in our case suggest steady performance across classes as well. The research highlights the strength of MLP in detecting malicious traffic.



```
Confusion Matrix:
 [[8914 3919]
 [ 458 9253]]
```

Figure 8: Confusion Matrix of MLP

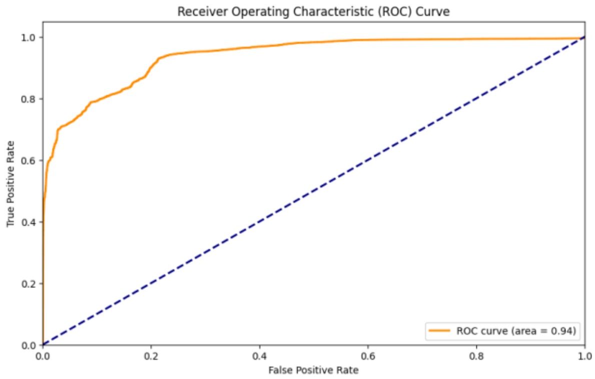


Figure 9: ROC Curve of MLP Neural Network

5.4 Data Visualization

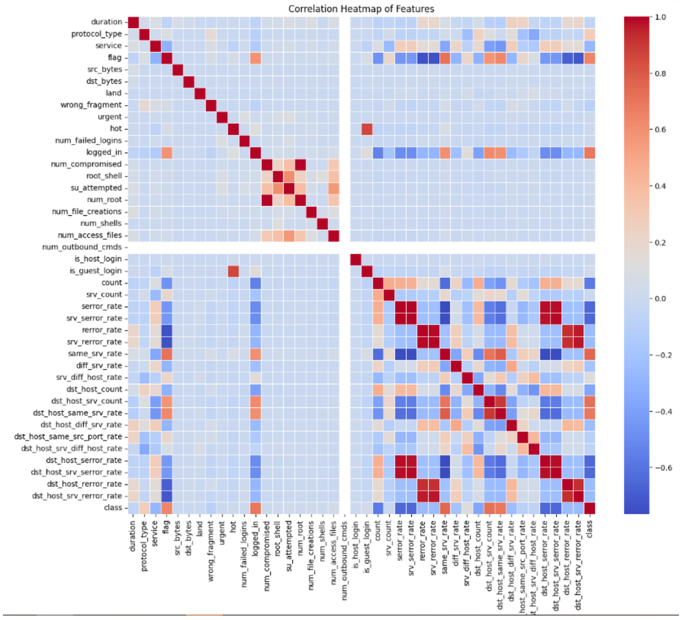


Figure 10: Correlation Heatmap of Numerical Features

The relationship of features of the dataset with each other is revealed in the correlation heatmap. Strong dependencies are demonstrated with strong correlations (red) whereas weak correlations (blue) depict independence. Some features such as protocol_type and dst_bytes are weakly associated while features like srv_serror_rate and serror_rate have strong positive correlation with each other. Based on the insights the features are selected and the models are tuned.

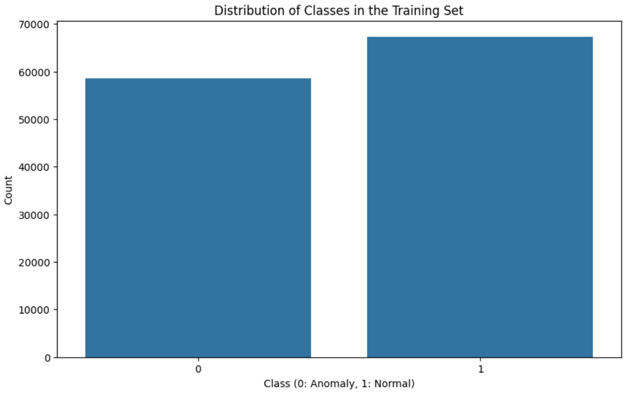


Figure 11: Distribution of Classes in the Training Set

6 Evaluation

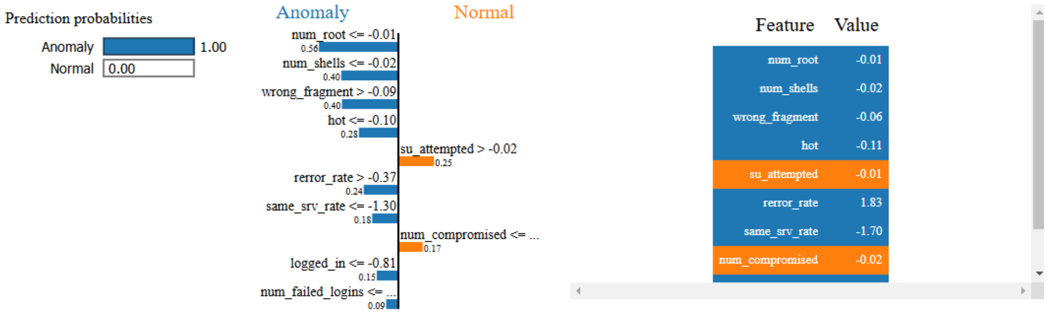
6.1 Comparative Analysis of Random Forest, SVM, and MLP classifier

Table 4: Comparative Analysis of Random Forest, SVM and Neural Network

Metric	Random Forest	SVM	MLP Classifier
F1-Score	0.79	0.82	0.81
Accuracy	0.79	0.82	0.81
Precision	0.93 (Class 0), 0.69 (Class 1)	0.90 (Class 0), 0.74 (Class 1)	0.95 (Class 0), 0.70 (Class 1)
Recall	0.68 (Class 0), 0.93 (Class 1)	0.76 (Class 0), 0.89 (Class 1)	0.69 (Class 0), 0.95 (Class 1)
Macro Avg	0.81	0.83	0.83
Weighted Avg	0.83	0.83	0.84

The comparison study demonstrated Random Forest, SVM, and MLP as the best classifiers while carrying out intrusion detection. For normal traffic, Random Forest records a high value of 93% accuracy and low value of 68% recall, thus obtaining a balanced F1-score of 0.79. However, SVM has a competitive F1-score of 0.82, higher recall at 89% for adversarial traffic metrics, which improves recall, thus maintaining SVM reliability. MLP classifier also had competitive values at 0.81 for F1 score showing trustworthy recall of 95% and accuracy of 95% for benign and adversary traffic respectively. SVM gives weight to all metrics therefore increases adaptability for intrusion detection while MLP gives solid accuracy and recall value on average.

6.2 XAI: LIME and SHAP



Feature	Value
num_root	-0.01
num_shells	-0.02
wrong_fragment	-0.06
hot	-0.11
su_attempted	-0.01
rerror_rate	1.83
same_srv_rate	-1.70
num_compromised	-0.02

Figure 12: Output of LIME

For a 1.0 likelihood anomaly prediction, LIME (Local Interpretable Model-agnostic Explanations) output explains this aspect. "Anomaly" (blue) and "Normal" (orange) are both pushed as valid during the prediction on the left panel. For example, the class 'Normal' is supported by information like su_attempted > -0.02 but due to num_root <= -0.01 and num_shells <= -0.02 anomalies can be predicted quite confidently. Such feature values, which are very influential for the decision on several choices including num_root and su_attempted, are depicted on the right panel.

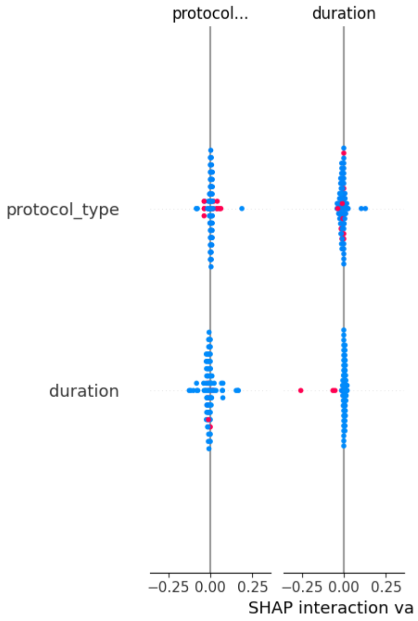


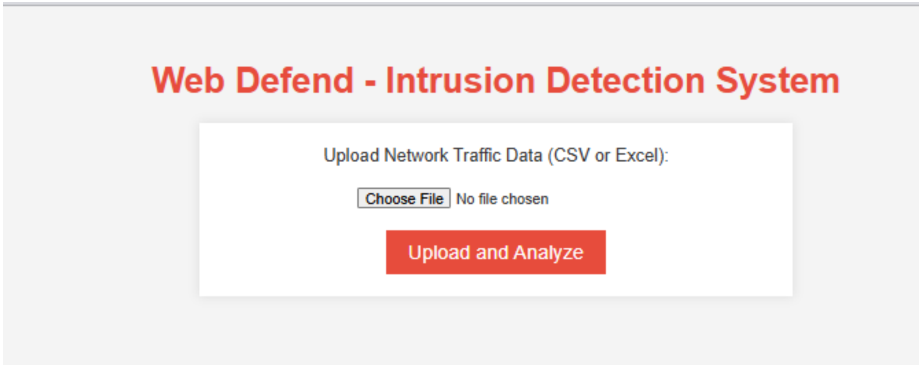
Figure 13: Output of SHAP

The SHAP (SHapley Additive exPlanations) depicts the influences of protocol_type and duration on the predictions made by the model in terms of the interaction with each other. The x-axis represents the SHAP interaction value where positive values increase the tendency for abnormality predictions and the negative, the utilization. Presented in the picture, the colour gradient presents the relative strength of features. Feature interactivity for protocol_type is more or less symmetrical around zero but time does exert preferentially more. This visualization explains feature interactivity and its influence on predictions which also makes the intrusion detection model understandable and key attributes easier to identify.

6.3 Error Analysis

In U2R and R2L assaults in particular, the feature distributions of legitimate and malicious traffic often overlapped, leading to prediction failures. The dataset is imbalanced and the model learning is constrained because these assaults are infrequent. Inaccuracies in the model's predictions could have been exacerbated by dataset noise and missing values.

6.4 GUI: PyCharm

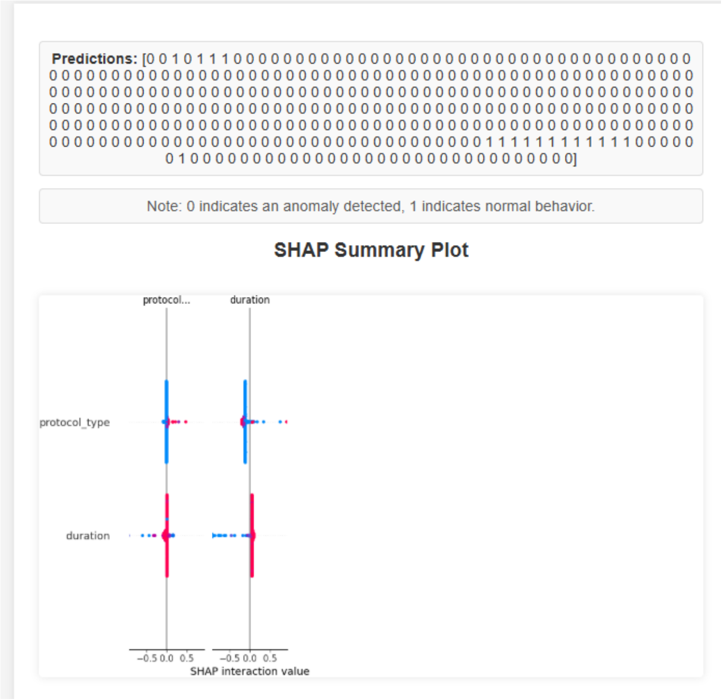


Web Defend - Intrusion Detection System

Upload Network Traffic Data (CSV or Excel):

Figure 14: Intrusion Detection System

This web application, created in PyCharm, allows everyone to load and manage network traffic in a CSV format. It is an Intrusion Detection System (IDS) interface. This makes it simple for the users as they can select the option of uploading a CSV which contains information regarding the network traffic. After a user has uploaded the appropriate file, the next step is to analyze the data by tapping the "Click to Detect Intrusion" button. For example, the submitted information may contain data that is processed by a backend model that could be a machine learning classifier for outlier detection or intrusion detection. The competent users are then given the results.



```
Predictions: [0 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0
          0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
```

Note: 0 indicates an anomaly detected, 1 indicates normal behavior.

SHAP Summary Plot

Figure 15: Intrusion Detection Result

7 Conclusion and Future Work

7.1 Conclusions

So as to enhance the usability of intrusion detection systems for the security analysts, the focus of this paper is on the integration of Explainable Artificial Intelligence methods such as SHAP and LIME into these systems. SVM outperformed the other classifiers in accuracy and recall when it came to detecting fraudulent traffic, whereas MLP performed better on average for major parameters. By combining interpretability with efficient detection, the study contributes in improving intrusion detection and paves way for more reliable and effective cyber security solutions. Still, there is the need for more improvement of feature interactions since the results indicate limitations though such as occasional misclassifications. This study underlines the potential of Explainable AI in improving the trust subjects have in these cyber security systems and points out that more work is needed to investigate the broader applications of it.

7.2 Future Works

To address the limitations of intrusion detection in determining this parameter to increase its accuracy even further, advanced deep learning models such as RNNs and CNNs will be necessary for future work. The constraints that the models have towards the evolving trends in cyber security threats can only be enhanced by adopting datasets that are more eclectic and current. The new attack vectors may be demystified by the application of unsupervised learning approaches. The use of integrated gradients or other tools can help refine the explainability of the models making the IDS system seen as more trustworthy and comprehensible when deployed for cybersecurity functions.

References

Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J. and Ahmad, F., 2021. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies, 32*(1), p.e4150.

Aldweesh, A., Derhab, A. and Emam, A.Z., 2020. Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues. *Knowledge-Based Systems*, *189*, p.105124.

Alzahrani, A.O. and Alenazi, M.J., 2021. Designing a network intrusion detection system based on machine learning for software defined networks. *Future Internet, 13*(5), p.111.

Ashiku, L. and Dagli, C., 2021. Network intrusion detection system using deep learning. *Procedia Computer Science*, *185*, pp.239-247.

Gamage, S. and Samarabandu, J., 2020. Deep learning methods in network intrusion detection: A survey and an objective comparison. *Journal of Network and Computer Applications*, *169*, p.102767.

Imrana, Y., Xiang, Y., Ali, L. and Abdul-Rauf, Z., 2021. A bidirectional LSTM deep learning approach for intrusion detection. *Expert Systems with Applications*, *185*, p.115524.

kaggle.com, 2023. *NSL-KDD99 Dataset.* [Online] Available at: https://www.kaggle.com/datasets/kaggleprollc/nsl-kdd99-dataset

Kilincer, I.F., Ertam, F. and Sengur, A., 2021. Machine learning methods for cyber security intrusion detection: Datasets and comparative study. *Computer Networks*, *188*, p.107840.

Mahbooba, B., Timilsina, M., Sahal, R. and Serrano, M., 2021. Explainable artificial intelligence (XAI) to enhance trust management in intrusion detection systems using decision tree model. *Complexity*, *2021*(1), p.6634811.

Mane, S. and Rao, D., 2021. Explaining network intrusion detection system using explainable AI framework. *arXiv preprint arXiv:2103.07110*.

Ozkan-Okay, M., Samet, R., Aslan, Ö. and Gupta, D., 2021. A comprehensive systematic literature review on intrusion detection systems. *IEEE Access*, *9*, pp.157727-157760.

Rabbani, M., Wang, Y., Khoshkangini, R., Jelodar, H., Zhao, R., Bagheri Baba Ahmadi, S. and Ayobi, S., 2021. A review on machine learning approaches for network malicious behavior detection in emerging technologies. *Entropy*, *23*(5), p.529.

Saltz, J.S., 2021, December. CRISP-DM for data science: strengths, weaknesses and potential next steps. In *2021 IEEE International Conference on Big Data (Big Data)* (pp. 2337-2344). IEEE.

Schröer, C., Kruse, F. and Gómez, J.M., 2021. A systematic literature review on applying CRISP-DM process model. *Procedia Computer Science*, *181*, pp.526-534.

Su, T., Sun, H., Zhu, J., Wang, S. and Li, Y., 2020. BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset. *IEEE Access*, *8*, pp.29575-29585.