National College of Ireland

Intrusion Detection Using Machine learning With Real-Time Dashboard

MSc Research Project
Data Analytics

Sangeetha Mora
Student ID: 23219602

School of Computing
National College of Ireland

Supervisor : Noel Cosgrave

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Sangeetha Mora

Student ID: 23219602

Programme: Data Analytics **Year:** 2024

Module: MSc Research Project

Supervisor: Noel Cosgrave
Submission Due Date: 11/12/2024

Project Title: Intrusion Detection Using Machine learning With Real-Time Dashboard

Word Count: 7756 **Page Count :** 22

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 bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Sangeetha Mora

Date: 11/12/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only

Signature:	
Date:	
Penalty Applied (if applicable):	

Intrusion Detection Using Machine learning With Real-Time Dashboard

Sangeetha Mora
23219602

Abstract

Every day that passes sees new, more complex, and developed cyber threats thus requiring strong IDS that offers real-time detection and analysis. This study aims to design an IDS based on machine learning to identify abnormal occurrences and directly display the result, which can effectively increase cybersecurity defence and improve operations.

The study employs the UNSW-NB15 dataset and first and second-level system and firewall data of an Ubuntu operating system-based virtual machine. In classification tasks, Decision Trees, Random Forests, and Gaussian Naive Bayes are used to classify the feature vectors appropriately. For the process of feature selection, Recursive Feature Elimination is applied whereas for anomaly detection in the user's authentication data Isolation Forest Algorithm is used. Logistic Regression is being used on firewall data for penetration prediction. Data preprocessing takes care of the appropriate arrangement of the gathered data and applying qualitative analysis.

According to the model, the highest accuracy of classification was 95.01% with the Random Forest model enabling measurement of high precision and recall. The analysis of the firewall logs using Logistic Regression yielded an accuracy of around 88.68%, with the F1 score touching the balance to ensure a better focus on the network intrusions. Some of the insights are fed into another real-time threat that will be constantly monitoring threats.

The proposed method fills the gap between formal and intuitive analysis of logs with current systems of machine learning in which the approach is not only scalable and efficient but adaptable to real-time intrusion objectives. Further improvements will target at including advanced technologies (e.g., deep learning), adding real-time alert sender, and these kinds of techniques to the distributed system and so forth.

Keywords: Decision Trees, Random Forests, Gaussian Naive Bayes, , Recursive Feature Elimination, firewall data, Authentication data, Intrusion Detection System.

1 Introduction

1.1 Overview

An intrusion detection system (IDS) is an important component of the cybersecurity framework due to its ability to watch traffic flow across a network and alert the appropriate parties to

potential intrusions. The research aims to develop a real-time intrusion detection system through machine learning techniques, with the UNSW-NB15 dataset, and fern simulation of authentication and firewall data in Ubuntu virtual machines. The UNSW-NB15 dataset, commonly used in intrusion detection, offers a mix of normal traffic and generated attack types as part of the ground truth, thus creating the most solid starting point for model training and evaluation.

The study uses a Decision Tree, Random Forest, Gaussian Naïve Bayes, and Recursive Feature Elimination algorithms to analyse the UNSW-NB15 dataset. Moreover, time-series anomaly detection is applied to the authentication data by using an Isolation forest while the Firewall data are analysed by using a logistic regression. These above-mentioned strategies are to increase the amount of accuracy and efficiency of intrusion detection regardless of the data types (Reza *et al.* 2021). Depending on the facts gathered from data, the paper organizes unstructured data to come up with important information, which balances the research in addressing current security issues.

1.2 Background of the Research

Being an inseparable part of modern security concepts and policies, intrusion detection systems are considered objective tools against the growing complexity of cyber risks. The static or traditional methods of identification of threats mainly based on signatures provide inadequate results, particularly concerning emerging threats in the system environment which proves the need for a smart and dynamic solution (Sah *et al.* 2024). Machine learning was also found to be a viable solution as it gives the possibility to investigate multifaceted structures and disorders signifying security threats.

UNSW-NB15 was developed by the Cyber Range Lab at UNSW Canberra and represents a realistic environment for studying IDSs and is used in this work. Accommodating both legitimate and intrusive behaviours, it is comprised of nine attacks and 49 attributes, which are useful in designing complex detection algorithms. During the syslog for each system consolidates these messages, adding authentication data and firewall data adds more structured data categories to the dataset's informational context, which allows for better anomaly detection.

Given this, the purpose of this study will be to try and narrow down the gap between sounding theoretical IDS frameworks and putting them into practice, which will include the use of both structured and unstructured data, real-time detection mechanisms, and real-time detailed information about the vulnerabilities in systems.

1.3 Aim and Objectives

Aim

The report aims to build an IDS in real-time using machine and deep learning algorithms with features from the UNSW-NB15 data set and system data to improve the detection and analysis of various cybersecurity threats with various algorithms that can help in determining appropriate anomalies and vulnerabilities successfully.

Objectives

- To analyse unstructured data of authentications and firewalls for further analytical purposes.
- For the usage and evaluation of Machine learning techniques such as Decision Tree, Random Forest, Gaussian Naïve Bayes, and Recursive Feature Elimination for intrusion detection techniques.
- To perform anomaly detection using time-series data on Isolation Forest on the authentication data.
- To create a live monitoring dashboard for the visualization of Intrusion Detection System results and reaction decision support.

1.4 Research Question

How effective are machine learning techniques in detecting and classifying intrusions within datasets, including network traffic data and system authentication & firewall data that have been pre-processed for analysis?

1.5 Research Rationale

- Through the recent increased frequency and complexity of cyberattacks, there is a need to develop improved real-time detection tools that are flexible enough to detect threats in different types of data. Traditional methods generally have trouble pointing to newly emerging or otherwise unknown patterns of attack. Built on the concept of statistical learning theory, machine learning provides an effective, data efficient, and reusable method of pattern recognition of outliers in large datasets. Combining set structured data, for instance, UNSW-NB15 with unstructured logs from real-world systems improves the possibility of applicability of the systems which falls under the gap between inherent theoretical solutions and real-life solutions.

1.6 Structure of the Report

The report is structured as follows:

- ***Chapter 2 : Introduction -*** This chapter presents an outline of what is to be covered as the overview, background, aim, objectives, research questions, rationale, and the organization of the report.
- ***Chapter 3 : Related work -*** Summarizes the current literature and techniques on intrusion detection systems with a focus on ML and real-time visualization.
- ***Chapter 4 : Research Methodology -*** Explains the process followed, data acquisition, data cleaning and normalization, dimensionality reduction and finally applying machine learning techniques.
- ***Chapter 5 : Design and Implementation Specifications -*** Explaining the development of the real-time intrusion detection system and the tool and technique used while developing it along with the dashboard.

- ***Chapter 6 : Evaluation -*** Reviews the results of machine learning models, considers the findings obtained from structured data, and compares the performance of the real-time dashboard.
- ***Chapter 7 : Conclusions and Discussion -*** The chapter concludes with a discussion of the limitations of the findings and specifies the directions of future research.

2 Related Work

The evolution of Intrusion Detection Systems (IDS) has been a significant one in the face of the ever-increasing complexity of cyber threats. Traditional Intrusion Detection System (IDS) approaches are the earliest where the focus of IDS is on signature-based detection techniques. Recent works focus on the contribution of ML and hybrid techniques to improve the IDS's efficiency and performance.

2.1 Intrusion Detection Systems (IDS) evolution

Traditional IDS methods, especially signature-based detection, depend significantly on established signatures for detecting threats. As note, these methods are effective against known attacks but will not anticipate novel or zero-day threats. Ahmad et al (2014) and Shrivastava (2017). Few approaches try to address these shortcomings are based on anomaly-based detection, which recognize deviations from the norm (Sheela et al., 2015; Yeo et al., 2017), but it results in high false-positive rates in real scenarios, making them less reliable. Some hybrid approaches have been developed that integrate misuse and anomaly detection, to improve accuracy and reduce false positives. Rajasekhar et al. (n.d.) and Al-mansor et al. (2018), hybrid systems can adapt to changing threats. But these systems face scalability and computational issues in large scale network environments (Elsadai et al., 2019).

This work expands upon these hybrid methods by incorporating both structured datasets and live analysis to enhance adaptation and magnitude, especially to address new threats occurring in dynamic network settings.

2.2 IDS: Machine Learning Techniques

The use of machine learning (ML) has revolutionized IDS by providing adaptable and scalable solutions. However, there are many popular ML techniques such as Decision Trees (DT), Random Forest (RF) and ensemble methods, which have been explored for their capabilities to detect intrusions. Decision Trees are relatively computationally lighter and provide interpretability along with quick decision making; however, they are susceptible to overfitting with large datasets (Mande et al., 2022; Reza et al., 2021). To overcome such limitations, Random Forest, a combination of several Decision Trees, is used to reduce overfitting and improve generalization (Sharma and Yadav, 2021; Mohamed and Agarwal, 2024).
This study employs Random Forest with Recursive Feature Elimination (RFE) for optimized feature selection, resulting in reduced computational overhead and enhanced model accuracy. RFE, gained specific mention by Sahu et al. (2020) and Mohamed and Agarwal (2024) The

most relevant features are selected by filtering the dataset to improve the performance of classification models.

Additionally, we apply Logistic Regression (LR), which is a well-established and interpretable tool, for firewall data analysis in classifying blocked IPs. Abbas et al. (2022) and Dini et al (2023) have discussed its applicability in intrusion detection systems. LR's generalization ability is proven to be very useful in real-world classification tasks (Kang et al. 2023) and therefore, it is an appropriate choice for the practical data analysis performed in this study.

2.3 The UNSW-NB15 Dataset: Strengths and Challenges

Data is a crucial part to training and assessing your IDS models. The UNSW-NB15 dataset is considered a contemporary benchmark for research in intrusion detection, containing various types of attacks and more realistic network traffic representations (UNSW, 2024). The UNSW-NB15 dataset was developed specifically to address the bugs found in older datasets such as KDD'99, through the issues of redundancy as well as stagnated attack pattern (Sharma and Yadav, 2021; Mohamed and Agarwal, 2024). But the dataset isn't without its challenges. Class imbalance and noisy features necessitate robust preprocessing techniques for normalization, outlier removal, and oversampling (Reza et al., 2021; Sah et al., 2024).

To overcome the UNSW-NB15 shortcoming and improve its classification accuracy, we utilize feature selection techniques including Recursive Feature Elimination (RFE) as a part of training and evaluating ML model on UNSW-NB15 dataset in this research. Dataset used for the actual experiments was presented with binary as well as multi-classification tasks, which fits the aims of this research very well.

2.4 Real-Time Network data Analysis

Though structured datasets are great for benchmarking, real-time data analysis is essential for practical IDS deployment. Data from the system (authentication data, firewall data) provide information on authentication attempts and firewall activities. The raw form of these data can be parsed by regex-based techniques to extract actionable features like timestamps of invalid login attempts and blocked IP addresses (Abbas et al., 2022; Kishore et al., 2024).

(As Ogah et al. 2024) note, machine learning models could be useful in detecting anomalous behaviour in the data. For example, it uses time-series analysis to identify anomalies in failed login attempts and trends in blocked IP addresses. We then apply Logistic Regression to classify instances in the firewall datasets, where its simplicity and speed allow for possible real-time classification (Abbas et al., 2022; Dini et al., 2023).

2.5 Limitations of Current Research

Although much progress has been made, there are still several limitations in IDS research:

High False Positives: Anomaly detection assumes that anomalous behaviour or activities are always malicious while benign activity is considered normal; this results in a high false positive rate, causing alert fatigue (Shrivastava, 2017; Yeo et al., 2017).

Overfitting in ML models: As for Decision Trees, although they are computationally inexpensive, they suffers from overfitting with large, complex high-dimensional datasets (Mande et al., 2022; Reza et al., 2021).

Scalability Issues: Real-time applications fail to efficiently utilize neural networks and other resource-intensive models (Kumar et al., 2023).

Bias on specific Dataset: Most ML models work only on specific dataset, which limits generalizability (Dini et al., 2023; Gebremariam et al., 2023).

Limited Real-Time Integration: Very few studies focus on integrating structured datasets with real-time data analysis, thereby overlooking the dynamism in today's networking environments (Abbas et al., 2022; Sah et al., 2024).

These gaps highlight the requirement for a well-crafted IDS framework that not only ensures accuracy but also maintains efficiency and adaptability in real-time conditions.

2.6 Summary and Justification

The literature review presented in this study emphasizes the importance of machine learning (ML) on the evolution of IDS and the challenges that remain, such as high false positives, overfitting and limited real time applicability. Although the UNSW-NB15 dataset is useful for developing IDS, it has its limitations that necessitate careful preprocessing and feature selection process. Moreover, computational data has been explored, but real-data computations are often left out of current research.

This thesis fills these gaps in two ways:

Using Decision Trees and Random Forest out of which Recursive Feature Elimination is applied to select our features while reducing Overfitting and improve Classification Performance.

Time-series anomaly detection for real time analysis of authentication data and firewall data to detect login anomalies and classify blocked IPs.

Illustrating the usefulness of Logistic Regression for application in real time at a classification task.
(Shameem et al.2024) suggested similar findings in their work, supporting and complementing these efforts. (Abbas et al. 2022), resulting in an important step towards improving the scalability and robustness of IDS solutions.

3 Research Methodology

3.1 Introduction

This paper describes the design of a hybrid ML-based IDS that combines structured datasets with real-time logs of the systems under observation. The main goal focused on providing a solution to all challenges, including false positive rates, overfitting, scalability, and what practical use IDS has in real-time. Using the UNSW-NB15 dataset[1] with authentication and firewall logs, a robust adaptive framework is developed in the methodology. This process involves data cleaning, feature engineering, model selection, and rigorous evaluation through metrics.

3.2 Research Approach

This research was derived from qualitative research approach for summarizing network activity and pattern recognition of intrusion detection in both network traffic and system data. As IDEASCALE (2023) describes, this method allows for the discovery of context-specific actions related to intrusion over heterogeneous datasets. Research Methodology Implemented in the Study The research methodology implemented in the study was segregated broadly into three phases which are data preprocessing, model building, and evaluation of the model. These phases allowed for an organized method of tackling theoretical issues whilst meeting practical needs for scalability and runtime effectiveness.

The orientation towards qualitative methods is consistent with recent work done by Dini et al. (2023) and Gebremariam et al. (2023), arguing that all intrusion data from the real world is best modelled and analysed within a contextual framework, reinforcing the need for flexible and adaptable analysis.

3.3 Research Methods

Data Preprocessing

Preprocessing involves preparing both structured and unstructured data for analysis. Therefore, the UNSW-NB15 dataset (obtained from UNSW Canberra Cyber Range Lab) was cleaned by eliminating the redundant features and normalizing it to achieve a common scale. Reza et al. discussed handling class imbalance using solutions including Synthetic Minority Oversampling Technique (SMOTE) (2021) (Mouhamed and Agarwal 2024). The dataset is split into training and testing subsets while maintaining the distribution of the different attack classes using stratified sampling.

Unstructured data coming from authentication and firewall logs were parsed and formatted into structured data. For authentication data, regex patterns were used to extract attributes like timestamp, username, event status, IP address, etc. Firewall logs were also structured in the

[1] https://research.unsw.edu.au/projects/unsw-nb15-dataset

form of features like src and dst IPs, protocols, ports, packet length, and flags (SYN, ACK, and the like). Statistical handling of missing entries was context dependent and done through either imputation or exclusion methods.

Feature engineering

Feature engineering was mainly about choosing and forming features which led to noticeable improvements of models. As per Sahu et al., Recursive Feature Elimination (RFE), was utilized to determine significant features, Mohamed and Agarwal (2024). A practical use their various classifiers: in the UNSW-NB15 dataset, to reduce computational overhead, irrelevant features were excluded, features with the most significant contribution to classification accuracy were retained.

Model Selection

A combination of supervised and unsupervised ML models was used in this study, which were selected as some of the most viable candidates for IDS applications. We chose to use Decision Trees due to their interpretability and computational efficiency, which aligns with finding of Mande et al. (2022). But their susceptibility to overfitting was alleviated using Random Forest as an ensemble method, which provides enhanced generalization and robustness (Sharma and Yadav, 2021). To classify probabilistically, the authors employed Gaussian Naïve Bayes, which is appropriate for "real time" regarding its simplicity (Gebremariam et al., 2023). According to the support of Abbas et al(2022) and Panwar (2022). Logistic Regression was selected due to its explainability and efficiency in predicting blocked IPs from firewall data.

Isolation Forest was also used to detect time-series anomalies by analysing authentication data and identifying anomalous login patterns. Ogah et al.(2024) highlighted how this algorithm was suitable in detecting anomalous behaviour and has the ability of handling high dimensional data as well as isolating the anomalies which made it a good choice for analysing system data in real-time.

Time-Series Anomaly Detection

A Time-series analysis was used to find anomalies in system data. We analysed authentication data over a rolling window to detect unusual spikes in failed login attempts. Isolation Forest tagged these as anomalies as potential brute-force attacks or other unauthorized login attempts. Likewise, in similar patterns were analysed in firewall data spikes of UFW BLOCK actions and Synchronize flag without Acknowledgment flag to identify Synchronize flood attack. These produced actionable insights with time-series plots and anomaly scores that made sure threats were detected in time.

3.4 Evaluation Metrics

The robustness and effectiveness of the proposed IDS models in terms of the detection of intrusions was evaluated using various performance metrics. Metrics such as accuracy,

precision, recall, F1 score, and AUC-ROC were used. Sah et al.(2024) noted that accuracy offered a general metric of correctness, while precision emphasized a lowering of false positives, which is a central consideration for making IDSs usable on a practical level in real life scenarios. Recall is what ensured that true intrusions were not missed, addressing the need for high detection rates highlighted by Dini et al.(2023). The F1 score provided a balance between precision and recall, resulting in a more holistic performance metric, while AUC-ROC assessed the discriminating performance of the model at various thresholds, especially helpful in imbalanced datasets.

These metrics were chosen for their applicability to the objectives of IDS and were computed per model to evaluate the performance from different perspectives. These findings were in keeping with the recommendations of Abbas et al. (2022) and Reza et al. (2021) on key trade-offs between detection rates and false positive suppression.

3.5 Data Collection

Employing the research based on secondary data collections, the project incorporates various types of data sets to make it more comprehensive on intrusions. The data used for this work is the UNSW-NB15 data set obtained from the UNSW Canberra Cyber Range Lab. The dataset contains a combination of actual current normal routines and simulated assault actions. It is made of raw network packets captured with tools and provides roughly 100 GB of traffic (UNSW, 2024). These are comma-separated values files encompassing over 2.5 million records and 49 features that capture Network and Traffic characteristics. Some of these features include attack types like Fussers, Dose, backdoors, and shell code, among others.
Other datasets consist of system authentication data (auth.log) and Firewall data (ufw.log). These include information about attempts to authenticate to the system and firewall data. Because of the distributed and unformatted structure of the data, certain changes had to be made pre-processing the data to get it in the right format for analysis. These data help in analysing the empirical behaviour of principals during logins and various attempts at unauthorized access that improve upon the IDS.
The datasets adopted in this research were selected because these covered different attack types and normal traffic which allowed the design of and testing of the machine learning algorithms under different simulated conditions. All data sets were analysed following data privacy and ethical guidelines.

3.6 Summary

The chapter described the processes followed in the development of an IDS based on a machine learning approach. The general idea of the approach was explained and, in the section, ‘research methods’ the specific steps that were taken, before building the models, such as preprocessing, feature selection, and model comparison were outlined. The data collection process was explained in detail and the authors wanted to collect secondary datasets and the UNSW-NB15 dataset and system data. Accommodation of multiple data sources enables the design of enhanced IDS that achieves real-time detection and monitoring of unauthorized intrusion.

4 Design Specification

4.1 Techniques

The research uses diverse algorithms from the field of machine learning and different data analysis techniques to define intrusion patterns and abnormalities in Network and System data. The techniques are divided primarily into three large groups depending on the input data and the type of analysis performed

4.1.1 Prediction of intrusion detection from the UNSW-NB15 dataset

The UNSW-NB15 dataset is used as the main source of network traffic flow. The following machine-learning algorithms and techniques were utilized:
Decision Tree Classifier **:** it is a supervised learning algorithm that builds a tree like structure by splitting a dataset on feature threshold. This effectively allows for interpretable decision-making, which is especially helpful for understanding traffic flow.
Random Forest Classifier **:** A composite model that makes decisions based on the input of multiple decision trees to enhance precision and minimize overfitting. Such approaches have shown great advantages for complicated datasets, like UNSW-NB15.
Gaussian Naive Bayes Classifier **:** A probabilistic model that uses Bayes' theorem and assumes that features are independent. It is cost-effective and perfect for real-time classification tasks.
Recursive Feature Elimination (RFE) : In this approach, we train the model iteratively and remove the least important features to rank and keep only essential features. Random Forest with RFE improves classification by selecting only the most informative features.

4.1.2 Detecting Anomalies in Authentication Data

Authentication data were analysed from Ubuntu systems to find abnormal attempts at login. The techniques used include:
Time-Series Anomaly Detection : Authentication failures were treated as time-series data and Isolation Forest was utilized to detect anomalies. It is an unsupervised algorithm that provides anomaly detection by exploring the lack of data while clustering.
Data Structuring and Insight Extraction : To prepare this data, we used regular expressions to parse raw data into structured formats. Data was extracted to inform models for anomaly detection by pulling out key trends, like failed logins, frequency, etc

4.1.3 Analysis of Firewall data

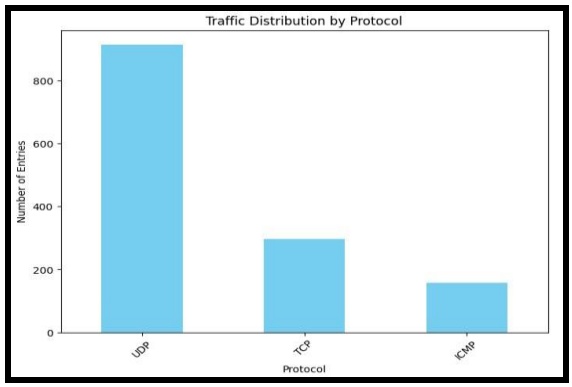

Figure 4.1.1: Traffic Distribution by Protocol

(Source: Self-Created)

Free open-source firewall Ubuntu provided logs for firewall data to schedule and identify evil traffic patterns and unauthorized access. The methodologies implemented include:

Logistic Regression: The form of a classification algorithm that describes how the observable features of data are related to the probability of occurrence of a particular class: for instance, normal or malicious traffic.

Data Transformation and Analysis: The initially accumulated unstructured data was then pre-processed and the data values that were derived were used to help in determining potential security threats.

General Processing and Checking Method for Data

Across all datasets, the following pre-processing techniques were applied:

Duplicate and Missing Data Checks: Made sure that the centre datasets were not cluttered with duplicate or limited records.

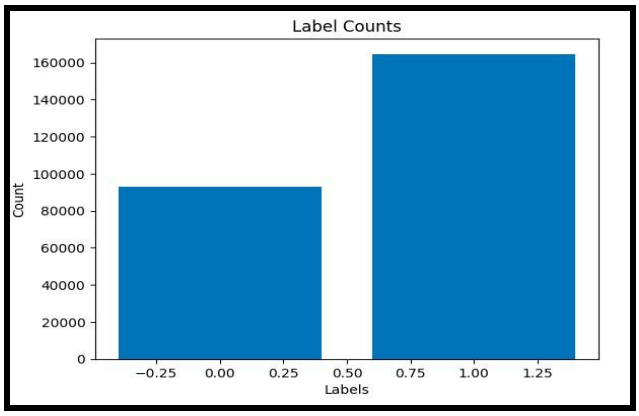

Figure 4.1.2: Label Counts

(Source: Self-Created)

Feature Encoding: Categories were transformed into numbers using Label Encoding due to compatibility with the algorithms.

Feature Scaling: Some numerical characteristics were scaled or transformed using such algorithms as StandardScaler and MinMaxScaler to improve the convergence of models.

Imbalance Handling: Applied resampling methods such as the Synthetic Minority Oversampling Technique (SMOTE), or under sampling to have a proper distribution of the minor classes in the UNSW-NB15 dataset.

Evaluation Metrics: The performance of the models was evaluated using parameters such as accuracy, precision, recall, F1-score, and AUC-ROC to possibly generalize the results of this assessment.

4.2 Framework

The overall framework of the project comprises of merging data sources, preprocessing pipelines, machine learning algorithms and a real-time dashboard. It is built to process data in a systematic way and provide actionable insights by the end of it.

4.2.1 Data Sources

The main sources of the data are as follows:

UNSW-NB15 Dataset : It is a very detailed dataset containing normal and malicious network traffic.

Authentication data **:** This captures an attempt to log into the system to help analyse any failed and/or anomalous access attempts.

Firewall data : Show what traffic was blocked or allowed through the firewall, useful for determining who did and did not get unauthorized access.

4.2.2 Data Preprocessing Pipeline

Here is the processing pipeline:

Understanding and Preprocessing : Data was loaded into Python environment like Jupyter Notebook, EDA to understand the design and content of data.

Feature Selection and Scaling : We applied RFE to target the critical features, and the various numerical attributes were scaled by techniques like MinMaxScaler to best suit the model training.

Creating a Structured Data : Respective data were tokenized and converted to a tabular structure, making them useful and ready for analysis.

4.2.3 Machine Learning Models

To bring these concepts to life, we divide the machine learning implementation into three tasks:

Network Traffic Analysis : Several classifiers including Decision Tree, Random Forest, and Gaussian Naive Bayes were used to classify network traffic as normal or malicious. Classification results significantly improved when using RFE, as only the most relevant features were retained.

Authentication Data : Isolation Forest detected abnormal activity in login attempts, indicating potential brute-force attacks.

Firewall Data : Logistic Regression was used to classify traffic events as legitimate access vs malicious access attempts.

Validation

The context gives model performance a broader perspective and various performance matrices such as Accuracy, Precision, Recall, and F1-score are applied. In order to overcome these issues, cross-validation is used, and AUC-ROC analysis is performed to compare relative model performance.

4.2.4 Real-Time Dashboard

The integration of the results into a real-time dashboard is useful since it provides usability. There are different means by which network traffic classifications, anomaly scores, and firewall analysis are displayed to support efficient monitoring of the intrusion events by the stakeholders (Gebremariam *et al.* 2023). Overall the structure outlined within this project guarantees the goals of the project are achieved and offers a solid and efficient solution for the real-time detection of intrusion.

5 Implementation

5.1 Outputs Summary

In the current research, the results demonstrate the realization and assessment of machine learning techniques on various datasets for intrusion detection purposes. Three distinct datasets were utilized: The first one is the UNSW-NB15 dataset, the second one is authentication data and the third one is firewall data. The primary instrument for data processing and machine learning was Python, using Scikit-learn, Pandas, NumPy, and Matplotlib libraries.

UNSW-NB15 Dataset Outputs

The decision tree, random forest, Gaussian Naive Bayes, and random forest with recursive feature elimination (RFE) were the machine learning models applied in this case on the UNSW-NB15 dataset. Figure 5.1.1: Summary of model performances including accuracy, precision, recall and training time The Random Forest achieved the highest accuracy of 95% and precision, as well as recall values were yielded as 96.22% and 95.96% respectively. The Random Forest model was optimized with Recursive Feature Elimination to minimize its computational overhead without compromising high model performance. The findings illustrate the model's ability to successfully classify network traffic as normal and anomalous Done

```
Performance Summary for All Models:
                            Model  Training Score  Accuracy  Precision  \
0         Decision Tree Classifier        0.997727  0.935875   0.951089
1         Random Forest Classifier        0.997727  0.950092   0.962241
2             Gaussian Naive Bayes        0.834214  0.835037   0.840620
3  Random Forest Classifier + RFE        0.995166  0.947427   0.958275

     Recall  Training Time (s)
0  0.948531          3.559590
1  0.959634         40.701515
2  0.915728          0.143990
3  0.959593        731.542675
```

Figure 5.1.1: Output Summary of the Designed ML Models

Authentication data (auth. log)

In case of the application structure authentication data used time-series analysis to detect anomalies because it makes this kind of data structured. After implementing Isolation Forest, it detected the times when too many failed logins happening, which can classify as a brute force attack. Output included anomaly scores, timestamps of anomalies, and visual trends showing

spikes in logins. Ultimately, the results of this process show how we can take unorganized data and organize it in a tabular form to provide insights that drive actionable decisions and detection of intrusions.

But the static threshold of Isolation Forest was an obstacle when handling a variety of intrusions. Future improvements may include dynamic thresholding or validation of our model using other different real world datasets.

Firewall data(ufw.log)

Pre-processed and structured data from firewall data was passed to Logistic Regression for analysis. The recalls and precisions of class labels from the classifier outputs are shown in Figure 5.1.2 which indicates the efficiency of the model. The Logistic Regression model shows an accuracy of 88.69% and a macro F1 of 73.00%, explaining a fair general performance. Based on the classification report and confusion matrix, it is evident that the model faced difficulty in detecting the minority class, i.e., 'malicious traffic', leading to low precision and recall for the same. Such inequalities are due to dataset imbalance and can be corrected through Synthetic Minority Oversampling Technique (SMOTE).

```
Class Weights: {0: 0.5321011673151751, 1: 8.287878787878787}

Classification Report with Class Weights:
              precision    recall  f1-score   support

           0       1.00      0.88      0.94       257
           1       0.35      1.00      0.52        17

    accuracy                           0.89       274
   macro avg       0.68      0.94      0.73       274
weighted avg       0.96      0.89      0.91       274


Confusion Matrix with Class Weights:
[[226  31]
 [  0  17]]

Accuracy Score with Class Weights:
0.8868613138686131
```

Figure 5.1.2 : Evaluation Metrics of Firewall data

All the output of these models forms a base for intrude detection efficiently. Each dataset reveals novel information useful for network traffic classification, failed user authentication patterns, and even firewall event classification; however, the identified limitations (i.e. imbalanced data; threshold sensitivity) indicate areas of future work for improvement.

5.2 Results and Critical Analysis

Evaluation of UNSW-NB15 Dataset Models

The Random Forest classifier performed better compared to other models and showed the highest accuracy, precision and recall of all tested algorithms. This enhances its efficacy for identifying network intrusions, specifically in tandem with Recursive Feature Elimination for efficient feature selection. The Decision Tree classifier performed slightly worse, and Gaussian Naive Bayes underperformed due to its inability to represent complex relationships amongst features. While RFE has shown to be highly effective, its high computational expense indicates that alternative, lower overhead feature selection methods should be explored.

Authentication data Insights

Isolation Forest works efficiently on authentication data to identify abnormal login attempts. This shows how unstructured data should be made structured to analyse them. Notably, the considerable dependence of the model on static thresholds makes it challenging to identify different kinds of intrusion. Dynamic thresholds and validation across supplementary datasets are a few future iterations that could lead to increased reliability.

Firewall data Analysis and Performance

Logistic Regression demonstrated acceptable results in distinguishing normal traffic in the firewall data analysis but struggled to classify malicious traffic, indicating that dataset imbalance is an issue. As depicted in Figure 5.1.2, the model produced high precision for the benign traffic, but had low recall for the minority class. Using resampling techniques such as SMOTE or ensemble methods would help balance this and improve classification performance.

Critical Insights

The results show that machine learning may effectively encompass various datasets for intrusion detection with some limitations including data imbalance, sparse data, efficacy in run time complex scenarios, etc. The obtained outputs establish a solid base for developing more capable IDS with improved preprocessing, hybrid approach, and integration into very time-sensitive applications. However, the model need to be improved and generalized further for adapting an IDS in dynamic and real world environments.

6 Evaluation

The evaluation section provides an in-depth analysis of what the results are from implementing machine learning techniques on the datasets. Finally, it is going over the specific experiment results, such as how good is the model doing in general, parameters, and invaluable accuracy benefits for potential network intrusion detection systems (NIDS). These experiments include the UNSW-NB15 dataset, firewall data received on routers and devices as well as authentication data, while sufficient analysis of each dataset is provided to determine how they fulfil the objective of the study.

6.1 Experiment 1: Decision Tree Classifier

In the current study, the Decision Tree Classifier was used on the UNSW-NB15 dataset for categorizing the network traffic as either normal or malicious. During training, the model was able to get up to an accuracy of 99.77%, meaning that it can pick out patterns from the given data. In the experiment, the accuracy of the evaluation in the test data reached 93.59% at the same time as the precision level and the recall level were 95.11% and 94.85% respectively. Such metrics demonstrate the high efficiency of the model to identify malicious traffic based on the parameters that are presented. Also, the training time was reasonably low, equal to 3.56 seconds, thereby providing for enhanced computational performance. Although the high training accuracy of the Decision Tree Classifier was impressive, the method was shown to have a degree of overfitting, when comparing the training accuracy to the test accuracy.

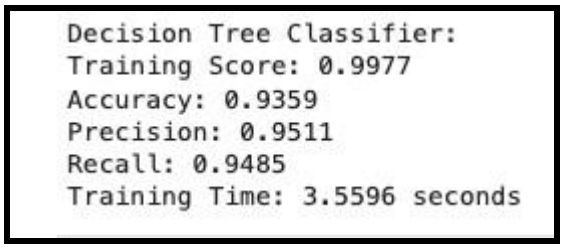

Figure 6.1.1: Evaluation Metrics of Decision Tree Classifier
(Source: Self-Created)

Figure 6.1.1: Evaluation Metrics of Decision Tree Classifier illustrates these results, highlighting the model's strengths and its limitation in generalization due to overfitting.

6.2 Experiment 2: Random Forest Classifier

To build upon the Decision Tree Classifier, the Random Forest Classifier was used with the help of an ensemble strategy. The applicative accuracy of the model was 99.77%, which also reflected the Decision Tree outcome. In the test data set, an increase in the accuracy to 95.01% was achieved, and the values inflated to 96.22% of the measure of precision and 95.96 % of the measure of recall. These results provide evidence of improved performance on unseen data and consistency over the Decision Tree Classifier. The time taken for training this model was relatively much higher, i.e., 40.70 sec the training time was high mainly due to the concept of ensemble learning. The increment in metrics of evaluation reveals that the Random Forest Classifier can perform well in cases of feature interactions and prevent overfitting. Figure 6.2.1: Evaluation Metrics of Random Forest
Classifier summarizes these findings, showcasing the superior performance of the Random Forest model on unseen data

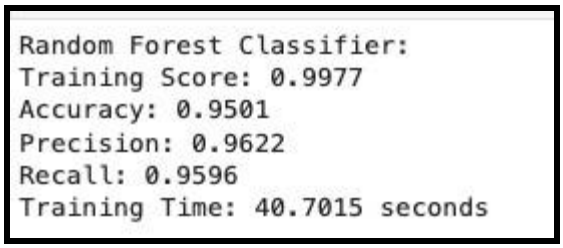

Figure 6.2.1: Evaluation Metrics of Random Forest Classifier
(Source: Self-Created)

6.3 Experiment 3: Gaussian Naive Bayes

Due to a benchmark method, the Gaussian Naive Bayes algorithm, a probabilistic classifier, was used. Training accuracy was 83.42%, and the test accuracy was 83.50%. The evaluated precision and recall measures were 84.06% and 91.57% respectively. As compared to the Decision Tree and Random Forest Classifiers, the model's performance rate was slightly lower but it was computationally effective with a training time of 0.14 seconds. Loss of accuracy was seen due to the low ability of the algorithm to deal with non-linear relationships and feature interactions. Also, it can take only one partition and since it is quite fast, it can be beneficial for

primary data analysis or low-resource setup. Figure 6.3.1: Evaluation Metrics of Gaussian Naive Bayes highlights the model's efficiency and its limitations in handling complex datasets.

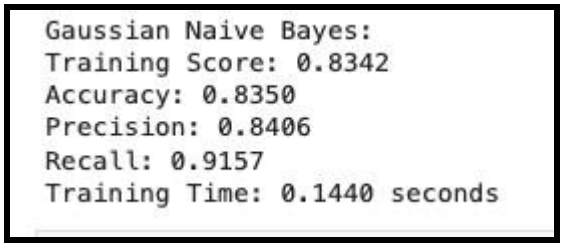

Figure 6.3.1: Evaluation Metrics of Gaussian Naive Bayes

(Source: Self-Created)

6.4 Experiment 4: Random Forest with Recursive Feature Elimination

The Random Forest Classifier was further enhanced using Recursive Feature Elimination (RFE) to know some of the best features. The approach reduced the overfitting of the model and across the board, the training accuracy stood at 99.52% while the test accuracy was 94.74%. Precision and recall results were also high at 95.82% and 95.95% respectively. The training time taken in this experiment was significantly higher and was 731.54 seconds because of the feature selection step. The analysis shows that the addition of the RFE step did help in the reduction of feature redundancy, and increased model interpretability, and overall performance while keeping modelling complexity at an optimal level. Figure 6.4.1: Evaluation Metrics of Random Forest with RFE illustrates the model's performance and the trade-off between computational cost and enhanced feature selection.

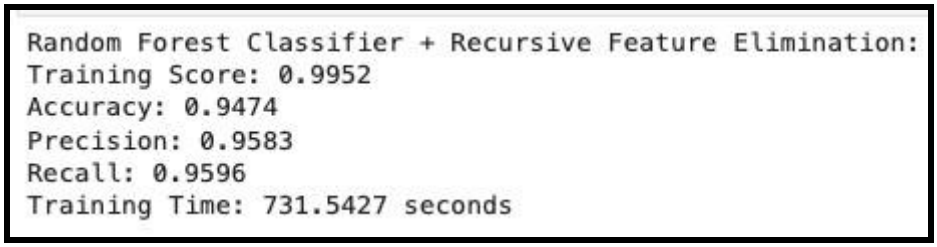

Figure 6.4.1: Evaluation Metrics of Random Forest with RFE

(Source: Self-Created)

6.5 Experiment 5: Firewall data Analysis with Logistic Regression

Firewall data were scaled, and then used for classification of the network traffic as benign or malicious using Logistic Regression. The overall accuracy of our model was 88.69% with a macro-average F1-score of 73.00%. The precision value for benign traffic (class 0) is high while precision and recall for malicious traffic (class 1) are lower, due to the imbalance in the dataset. To resolve this problem, methods like Synthetic Minority Oversampling (SMOTE) can be used. Figure 6.5.1: Evaluation Metrics of Firewall data presents the classification report and confusion matrix, highlighting the imbalance issue and its impact on performance.

```
Class Weights: {0: 0.5321011673151751, 1: 8.287878787878787}

Classification Report with Class Weights:
              precision    recall  f1-score   support

           0       1.00      0.88      0.94       257
           1       0.35      1.00      0.52        17

    accuracy                           0.89       274
   macro avg       0.68      0.94      0.73       274
weighted avg       0.96      0.89      0.91       274


Confusion Matrix with Class Weights:
[[226  31]
 [  0  17]]

Accuracy Score with Class Weights:
0.8868613138686131
```

Figure 6.5.1: Evaluation Metrics of Firewall data
(Source: Self-Created)

6.6 Experiment 6: Authentication data with Isolation Forest

To detect anomalies, we applied the Isolation Forest algorithm to the authentication data. The model detected high-volume failed login attempts and marked them as possible brute-force attacks. To visualize these anomalies, we used time-series analysis and found that Isolation Forest successfully identifies unusual patterns. Nonetheless, the algorithm's one-size-fits-all threshold restricted its ability to accommodate different intrusion types

6.7 Discussion

Therefore, the performance analysis of several machine learning models for intrusion detection shows the difference in their characteristics such as accuracy and time complexity etc. The Decision Tree Classifier had good classification and precision at low computational cost, but signs of overfitting. Random Forest Classifier generalization was enhanced and produced higher accuracy and recall although it took longer time to train. The Gaussian Naive Bayes model gave fair performance and computational speed but is inadequate in handling multi-way feature interaction.

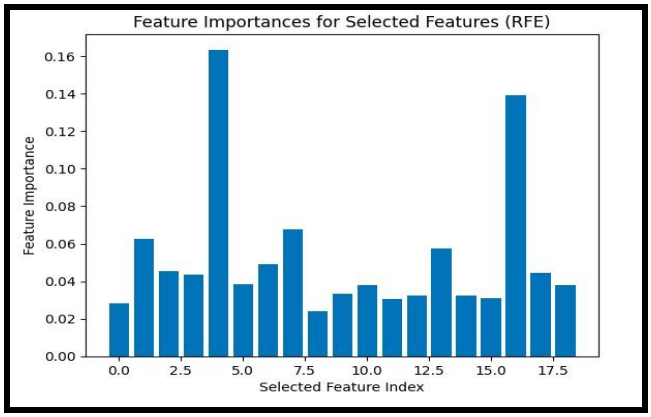

Figure 6.7.1: Feature Importance (Source: Self-Created)

Time complexity and flexible accuracy are the main differences between the methods when the Recursive Feature Elimination algorithm is combined with the Random Forest Classifier the performance and feature selection are improved while interpretability is added at the cost of longer training time.

After applying logistic regression on the firewall data there was a prediction imbalance for the classes and thus the recommendation for further optimization of the unstructured data. The entire study signifies the significance of feature extraction, suitable model selection, and data representational imbalance solutions for designing reliable and real-time intrusion detection systems.

7 Conclusion and Future Work

7.1 Linking to Objectives

The objective of this research study was to investigate and assess the success of machine learning methods for both detection and classification of intrusions within both datasets (network traffic data and refined system data). The objective of this research was accomplished through the implementation of Decision Tree, Random Forest, Gaussian Naïve Bayes, Recursive Feature Elimination and used time series anomaly detection with Isolation Forest. Steps of preprocessing which have structured the raw data in analysable formats and allowed the models in producing accurate results and actionable insights. Moreover, a framework for a real-time dashboard was implemented, solving the requirements for real world applications for intrusion detection.

7.2 Connection to Research Questions

The study proved that ML methods are suitable to extract the patterns of intrusions from various datasets. The UNSW-NB15 dataset facilitated high precision and recall for malicious network traffic detection, and a time-series anomaly detection approach on authentication data uncovered significant abnormalities in failed login attempts. There, Logistic Regression was still the most successful classifier for the firewall events, although it struggled with class imbalance. The results confirm the capability of ML models to generalize across heterogeneous data sources catering to objectives of intrusion detection.

7.3 Future Work

In addition, future research could broaden the real-time dashboard using dynamic visualizations/advanced alerting systems to enhance usability and responsiveness. Incorporating various data sources like IoT data and cloud traffic can improve the generalizability of the models. More sophisticated anomaly detection and feature extraction methods, such as RNNs and CNNs, have yet to be considered. Furthermore, using tools such as Apache Kafka and Spark allows for real-time processing of massive-scale datasets. Imbalanced datasets can be improved using techniques such as SMOTE and ensemble methods, which are crucial for enhancing classification outcomes in minority classes in firewall data.

7.4 Conclusion

Clearly, the work effectively proved the potential of use of machine learning algorithms in intrusion detection based on classical datasets and filtered data. Several models provided very

high accuracy (including Decision Tree, Random Forests, Gaussian Naïve Bayes) and Recursive Feature Elimination reduced feature redundancy and improved the selection of features. The application of the Isolation Forest helped for identifying the anomalies in authentication data which translated to the best coverage for the real time detection. The approach suggests that neural networks can be a promising avenue for scalable intrusion detection, despite some challenges like imbalance in dataset and complexity in computation. Next-generation visualization, data unification, and adaptive modelling play an important role to meet changing cybersecurity challenges.

References

Abbas, A., Khan, M.A., Latif, S., Ajaz, M., Shah, A.A. and Ahmad, J. (2022) 'A new ensemble-based intrusion detection system for internet of things', *Arabian Journal for Science and Engineering*, pp. 1–15.

Dini, P., Elhanashi, A., Begni, A., Saponara, S., Zheng, Q. and Gasmi, K. (2023) 'Overview on intrusion detection systems design exploiting machine learning for networking cybersecurity', *Applied Sciences*, 13(13), p. 7507.

Gebremariam, G.G., Panda, J. and Indu, S. (2023) 'Design of advanced intrusion detection systems based on hybrid machine learning techniques in hierarchically wireless sensor networks', *Connection Science*, 35(1), p. 2246703.

IDEASCALE (2023) *What is qualitative research? Definition, types, examples, methods, and best practices*. Available at: https://ideascale.com/blog/what-is-qualitative-research/ (Accessed: 2 December 2024).

Ogah, M.D., Essien, J., Ogharandukun, M. and Abdullahi, M. (2024) 'Machine learning models for heterogenous network security anomaly detection', *Journal of Computer and Communications*, 12(6), pp. 38–58.

Sah, G., Singh, S. and Banerjee, S. (2024) 'Intrusion detection system using classification algorithms with feature selection mechanism over real-time data traffic', *China Communications*, 21(9), pp. 292–320.

Panwar, N. (2022) *Anomaly infiltration detection in networks using machine learning*. **Mande, S., Ramachandran, N., Kumar, C.K. and Priyanka, C.N.** (2022) 'A brief analysis on machine learning classifiers for intrusion detection to enhance network security', in *Proceedings of the 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS)*, December, pp. 567–573. IEEE.

Sahu, A., Mao, Z., Davis, K. and Goulart, A.E. (2020) 'Data processing and model selection for machine learning-based network intrusion detection', in *Proceedings of the 2020 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*, May, pp. 1–6. IEEE.

Sharma, N. and Yadav, N.S. (2021) 'Ensemble learning based classification of UNSWNB15 dataset using exploratory data analysis', in *Proceedings of the 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, September, pp. 1–7. IEEE.

Reza, M.N., Kabir, S.F., Jahan, N. and Islam, M. (2021) 'Evaluation of machine learning algorithms using feature selection methods for network intrusion detection systems', in *Proceedings of the 2021 5th International Conference on Electrical Information and Communication Technology (EICT)*, December, pp. 1–6. IEEE.

Al Ahasan, M.A., Hu, M. and Shahriar, N. (2023) 'Ofmcdm/irf: A phishing website detection model based on optimized fuzzy multi-criteria decision-making and improved random forest', in *Proceedings of the 2023 Silicon Valley Cybersecurity Conference (SVCC)*, May, pp. 1–8. IEEE.

Mohamed, F.O.A. and Agarwal, M. (2024) 'Using Recursive Feature Elimination Feature Selection based Machine Learning Classifier for Attack Classification on UNSW-NB15 dataset', in *Proceedings of the 2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, April, pp. 1–7. IEEE.

Ahmad, H., Uppal, M. and Javed, M. (2014) 'An overview of intrusion detection system (IDS) along with its commonly used techniques and classifications'.

Shrivastava, U. (2017) 'A review of intrusion detection system'.

Yeo, L.H., Che, X. and Lakkaraju, S. (2017) 'Understanding modern intrusion detection systems: A survey'.

Sheela, S.N., Prasad, E., Srinath, M. and Basha, M.S. (2015) 'Intrusion detection systems, tools and techniques – An overview'.

Al-mansor, M.J., Beng, K. and Gan, K. (2018) 'Intrusion detection systems: Principles and perspectives'.

Rajasekhar, K., Babu, B.S., Prasanna, P., Lavanya, D. and Krishna, T.V. (n.d.) 'An overview of intrusion detection system strategies and issues'.

Azhagiri, M., Rajesh, A. and Karthik, S. (2015) 'Intrusion detection and prevention system: Technologies and challenges'.

Elsadai, A., Ibrahim, J., Hajjaj, F. and Jakić, P. (2019) 'The overview of intrusion detection system methods and techniques', *SINTEZA*.

Kishore, P.H., Rahiman, S.M., Mahidhar, P., Chandol, M.K. and Mahendra, T. (2024) 'Securing networks in the digital age: A review of intrusion detection and prevention strategies', *International Journal of Innovative Science and Research Technology*.

Kumar, A., Kumar, A., Singh, M. and Kumari, P. (2023) 'Cyber attack detection using deep learning', *Middle East Research Journal of Engineering and Technology*. **Kulkarni, S.** (2020) 'A survey on intrusion detection systems (IDS)'.