Enhancing Cybersecurity through AI-Driven Threat Detection Systems

MSc Research Project
Cyber Security

Harini Srinivasulu
Student ID :23187921

School of Computing
National College of Ireland

Supervisor: Prof.Eugene Mclaughlin

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Harini Srinivasulu
Student ID:	23187921
Programme:	MSc in Cyber Security
Year:	2024
Module:	MSc Research Project
Supervisor:	Eugene Mclaughlin
Submission Due Date:	12/12/2024
Project Title:	Enhancing Cybersecurity through AI-Driven Threat Detection Systems
Word Count:	7254
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:	Harini
Date:	11th December 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):	

Enhancing Cybersecurity through AI-Driven Threat Detection Systems

Harini Srinivasulu
23187921

Abstract

As organizations increasingly depending on the digital infrastructures, the complexity and frequency of cybersecurity threats have grown significantly. These new threats are hard to tackle by traditional security mechanisms, thereby jeopardizing the security of important systems. In response to this challenge, AI and ML have become innovative solutions in cybersecurity, which provide timely solutions for threat identification. This research study aims to evaluate the performance of the five different algorithms of machine learning, namely K-Nearest Neighbors (KNN), Decision Trees, Logistic Regression, Random Forest, and Deep Neural Networks (DNN) in identifying DDoS attacks from the network traffic data. Through the performance analysis of these models, it is also possible to compare the effectiveness of the different approaches to traffic classification with the goal of identifying the best approach for the identification of traffic type as benign or DDos malicious. The findings indicates that the proposed Deep Neural Network model provides the highest accuracy of 99.92% & is 100% precise, recall, and F1score for both classes. The Random Forest model also had high accuracy of 98.26% while KNN, Decision Tree, and Logistic Regression models had accuracy of 95.67% – 96.94%. These results emphasize the possibility of the application of AI-based systems for the faster and more accurate identification of cyber threats compared to conventional techniques. The study shows how which machine learning model to use for threat detection and how AI can be used for the prevention of threats in digital environments. As for the limitations of this study, future work will be devoted to improving the real-time detection rates and investigating the possibility of detecting multiple classes of attacks, which will improve the existing state of affairs in the sphere of cybersecurity.

1 Introduction

1.1 Background

The advancement in technology in the digital age has brought about dramatic changes that have affected how organisations manage themselves, how they interact with others and how they deliver services. However, this progress comes with a steep price: a rising threat environment in cyberspace that is both multifaceted and more precarious. Hackers constantly take advantage of the gaps in computer systems, using various strategies to infiltrate and disrupt information confidentiality, access, and reliability. Such security measures as the firewalls, intrusion detection systems, and antivirus programs are still relevant but are unable to provide adequate protection against new forms of cyber threats. Static rule-based system do not have the flexibility needed to mitigate against new and complex threats, putting organizations at great risk of financial, operational, and reputational losses.

To this increasing threat, AI and ML have risen as disruptive technologies in cybersecurity. Such technologies provide opportunities to process the excessive amounts of information in real mode, look for changes and complex patterns that indicate malicious actions. AI based threat detection systems on the other hand are not set and rigid and can analyse data from previous attacks and adjust to the new ways that the attackers may use to attack. Such systems help to detect and analyse threats automatically, which will greatly reduce the time needed to notice an attack and prevent damage.

This research will seek to establish the effectiveness of AI based systems in identifying and categorizing cyber-attacks in network traffic. Using Decision Trees, Logistic Regression, K-Nearest Neighbors (KNN), Random Forest, and Deep Neural Networks, this research aims to establish the best techniques in threat detection. One major concern is the system's ability to make the comparison between the legitimate and illegitimate type of network traffic, especially in the context of DDoS type of network traffic attacks. The ultimate objective is to design and implement the strong and flexible cybersecurity systems which can prevent threats & improve the overall cyber defences.

1.2 Motivation

This research has been motivated by the urgent need to overcome the shortcomings of conventional security approaches to cyber threats. There is not only an increase in cyber-attacks but also an increase in sophistication of attacks like polymorphic attacks, encrypted attacks and multi-vector attacks. It is not rare to see organizations caught in a defensive mode, trying to address risks, which have already penetrated traditional security perimeters.

AI provides a radical solution to the problem, as it becomes possible to act preventively in the sphere of cybersecurity. Artificial intelligence can take large amounts of data then analyse it to reveal the complexity of patterns and variations that are characteristic of cyber threats. This capability is very important in the situation where early detection and quick reaction is impossible. Studying the capabilities of AI-driven threat detection systems for improving the detection accuracy, speed and increasing the system's scalability are the main driving forces of this research to analyse the ways of the further enhancement of the AI systems to meet the current needs of the cybersecurity. Through the development of these technologies, this research aims at providing safer environment for organizations and people in the cyberspace.

1.3 Research Questions

To proceed with this study, the following research questions have been formulated:

- How effective are machine learning algorithms in detecting various types of cyberattacks, such as DDoS, within network traffic data?

  - Which machine learning model provides the best classification accuracy and performance in identifying cybersecurity threats?

  - Can feature engineering and data preprocessing techniques enhance the predictive capabilities of machine learning models in threat detection?

- What is the major impact of these AI-driven threat detection systems on the speed and accuracy of identifying potential cybersecurity threats compared to traditional methods?

These questions aim to covers the intersection of AI and cybersecurity, handling the key aspects of adaptability, accuracy, and practical implementation.

1.4 Research Objectives

The mains objectives of this research study are:

- To design and implement an AI-driven threat detection system that have ability to classify accurately the various types of cyberattacks within network traffic data. • To evaluate the performance values of the different machine learning algorithms, including Decision Trees, KNN, Logistic Regression, Random Forest, and Deep Neural Networks, in differentiating between the benign and DDos type of malicious network traffic.

- To identify the most effective algorithm or combination of algorithms for building robust and scalable threat detection systems.

- To assess the influence of the feature engineering & preprocessing methods on enhancing model performance.

1.5 Structure of the Research

This research study is systematically structured to achieve its objectives and provide a comprehensive understanding of AI-driven threat detection systems.

- Section 2: Literature Review provides an in-depth analysis of existing AI-based threat detection systems, their evolution, and the challenges associated with their deployment.

- Section 3: Methodology outlines the research design, including data collection, preprocessing, exploratory data analysis, and the training and evaluation of various machine learning models.

- Section 4: Results and Discussion presents the experimental outcomes, comparing model performance in various types of metrics such as the accuracy, recall, precision and F1-score. This section also discusses the implications of the findings in the context of cybersecurity.

- Section 5: Lastly, Conclusion & Future Work section summarizes the research study findings, highlights their significance, and proposes directions for future research in AI-driven threat detection.

In conclusion, the cyber threats are becoming more and more advanced and the approach of traditional static security models is no longer effective. AI-based threat identification systems are an emerging solution that can utilize machine learning algorithms to detect threats and prevent them with unparalleled efficiency and precision. The goal of this research is to enhance the state of cybersecurity by identifying if these technologies can distinguish and identify different types of cyber threats, which in turn will help create a better and safer cyber environment.

2 Literature Review

This research study objectives to analyze the state of study in the area of AI for cybersecurity, and the use of the ML approach for threat detection. This section is divided into four sections: an overview of cybersecurity threats and conventional security measures, AI-Driven Threat Detection Systems, Machine learning and AI in cyber security, limitations of AI based systems, new developments in threat detection using machine learning.

2.1 Overview of Cybersecurity Threats and Traditional Defense Mechanisms

The cybersecurity threats are grouped into malware, phishing, ransomware, DDoS, and APT. Data suggest that threats have evolved not only in quantity but also in quality and have been using sophisticated methods to avoid detection (Mallick and Nath; 2024; Mohan et al.; 2022). The traditional ways of implementing the security solutions are based on the IDS that is based on a signature, firewalls, and antivirus solutions that work based on a set of rules. Despite the capability of identifying known threats, these approaches are lacking in identifying zero-day vulnerabilities and failure to address new and complex attack methodologies are inefficient for the current generation security requirements (Mohamed et al.; 2024).

2.2 AI-Driven Threat Detection Systems

The threat detection systems that utilize AI are based on machine learning to prevent cyber threats actively. Such systems can identify different sorts of cyber threats such as malware, phishing, and the DoS attack by inspecting the network traffic, logs, and endpoints' interactions (Alhajjar et al.; 2021). Distributed Denial-of-Service (DDoS) attacks, by investigating the network traffic, logs, and endpoint interactions (Alhajjar et al.; 2021). There are different methods that are used in AI threat detection with reference to the case of anomaly-based detection where the normal system behavior is first determined, and deviations are then searched for to signify an attack (Skopik et al.; 2022). Anomaly detection is especially useful where new or unknown threats are expected since it does not use the signature of known attacks. The other type is the signature-based type that involves looking for specific attacks within the network traffic flow. Whereas, the signature-based methods are quite efficient in identifying known threats, it lacks efficiency while dealing with new or emerging threat types (Kothamali and Banik; 2022). There is also a combination of the anomaly-based and the signature-based approaches that has been attempted with the aim of increasing the accuracy and the reliability of threat detection systems (Agoramoorthy et al.; 2023).

2.3 Machine Learning Models for Threat Detection

Several algorithms of machine learning have been used for cybersecurity in order to detect and categorize cyber threats Some of these algorithms include KNN that is a simple algorithm for classification. This is done by the process of finding the closest distances to the data points in the training dataset and then labelling the data based on the voting system (Goodfellow et al.;

2016). KNN has been applied in anomaly detection problems, that is, problems involving the discovery of intruders in traffic. Another common algorithm used in cybersecurity is called Decision Trees, which analyses the data by dividing the features based on their values to form a tree like structure for classification (Almomani et al.; 2021). Decision Trees are especially preferred because of their good interpretability, as it is possible to trace the decision-making process made by them. However, they can be sensitive to over fitting, most especially when the data set is noisy. The Random Forest a classification technique that uses multiple decision trees have been reported to enhance classification performance by reducing overfitting (Ahmad, Rasool, Javed, Baker and Jalil; 2021; Sharma and Sharma; n.d.). Another popular technique is logistic regression which can be used for binary classification problems, for instance, to determine whether a connection is a DDoS attack or a phishing attempt, finding the likelihood of the input belonging to a class (Wiafe et al.; 2020). Last but not the least; Deep Neural Networks (DNNs) that come under the broad domain of deep learning are capable of learning both, feature and pattern hierarchy of a large dataset which makes them particularly suitable for identifying sophisticated threats such as malware and ransomware (Kavitha and Thejas; 2024; Sarker et al.; 2021).

2.4 Challenges in Implementing AI-Based Systems for Threat Detection

However, there are several issues that make the adoption of AI-based systems for cybersecurity challenging. Some of the issues include One of the most common issues in cybersecurity datasets is that the data is not equally divided between the normal and anomalous flows, it is highly skewed. This imbalance means that the models tend to focus more on identifying benign traffic, which leads to false negative and thus missing on threats (Pavithra and Vikas; 2024). To this end, oversampling, undersampling, and cost sensitive learning strategies have been suggested (Hasib et al.; 2022). The fourth threat is the requirement for constant identification since cyber threats can progress quickly. AI systems need to be ready to handle a vast amount of data and provide the prediction in real-time to be useful in eliminating threats. The issues of high computational time and latency that accompany real-time analysis of large datasets are the key problems of AI-based threat detection (Cadet et al.; 2024). Furthermore, machine learning models have another problem called adversarial attacks, which are input data modifications made with the specific aim of provoking wrong outputs from the AI system (Mensah and Boateng; 2024). This vulnerability is a big threat and even more so in high-risk areas such as national security or critical infrastructure. Last, interpretability is still a challenge, as most of the existing machine learning models, specially the deep learning networks, are referred as **black boxes.** This lack of interpretability makes it challenging for cybersecurity professionals to grasp how exactly a model came up with a given decision – which in turn makes it hard to foster trust and utilization of AI systems in security-sensitive contexts (Rahaman et al.; 2024).

2.5 Conclusion

The use of AI based threat detection systems is a powerful approach to improving cybersecurity because it can speed up the detection of threats. Main machine learning algorithms used for this purpose are Decision Trees, KNN, Logistic Regression, Random Forests, and Deep Neural Networks where these methods have shown a potential of providing solutions for many cyberattacks. However, there exist several limitations to the application of AI in cybersecurity, which include imbalanced datasets, real-time performance, adversarial attacks, and model interpretability. The results of this study suggest that there are still limitations to AI-driven threat detection and the following recommendations should be made for future research: Robustness, scalability, and interpretability of the threat detection models should be enhanced to overcome the mentioned challenges and guarantee the practical applicability of the systems.

3 Methodology

The aim of this work is to improve cybersecurity by deploying AI-based threat identification, with an emphasis on network intrusion attack classification. The study seeks to develop and test various machine learning algorithms to classify and recognise cyber threats including the DDoS attacks and other normal network traffic. This is done using a systematic process that comprises of data exploration, data preprocessing, feature selection, model building and model assessment.

3.1 Research Design

The research design is structured into several stages:

- Data Overview & Exploration: Gathering of relevant network traffic datasets, particularly those containing both benign and malicious traffic, such as DDoS attacks. • Data Preprocessing & Feature Engineering: Cleaning, transforming, and normalizing the data for use in machine learning models. Identifying and creating relevant feature attributes from the raw network intrusion traffic data to enhance the performance of machine learning models.

- Model Selection: Evaluating multiple machine learning algorithms including Decision Trees, K-Nearest Neighbors (KNN), Logistic Regression, Random Forest, and Deep Neural Networks (DNN).

- Model Training & Evaluation: Training the models on the prepared dataset and evaluating their performance using various metrics.

- Comparison of Results: Analysing and comparing the results of different models, discussing their strengths and weaknesses.

3.2 Dataset Overview & Exploration

For this research study, Employed the Network Intrusion Dataset, namely the FridayWorkingHours-Afternoon-DDos.pcap ISCX.csv datafile, which provides records on network traffic and different types of cyberattacks. This dataset is also available on Kaggle and contains traffic flows that are identified as BENIGN or belonging to other types of attacks including DDoS.

Dataset Overview:

- Total Entries: 225,745 rows (network traffic instances)

- Total Features: 79 columns describing various characteristics of the network traffic Table 1: Description of Key Columns in Network Intrusion Dataset

Category	**Column Name**	**Description**
Traffic Features	Destination Port	The port number for the destination of the traffic.
	Flow Duration	Duration of the flow in microseconds.
	Packet Lengths	Several features related to packet sizes, such as: - Total and individual forward (Fwd) and backward (Bwd) packet lengths. - Maximum, minimum, mean, and standard deviation of packet lengths.
	Packet Rates	Flow rate measurements like Flow Bytes per second (Flow Bytes/s), Flow Packets per second (Flow Packets/s), and various Inter-arrival Times (IAT) of packets.
Flag Features	Flags	Indicating the state of the TCP connection, such as FIN, SYN, RST, PSH, ACK, URG, CWE, and ECE flags. These flags are used in various protocols (e.g., TCP) and indicate specific control actions.
	Flag Counts	Count of each TCP flag type encountered in the flow.
Flow and Session Stats	Bytes and Packets	The number of packets and the total bytes exchanged during the flow, for both forward and backward traffic.
	Segment Size	Average and standard deviation of segment sizes for both forward and backward traffic.
Timebased Features	Active/Idle Time	Various metrics regarding the active time and idle time of the flow, including mean, standard deviation, max, and min.
	Time-based Counts	Counts of various flags and packets during certain time intervals.
Label	Label	Indicates the type of traffic: BENIGN (normal, non-malicious) or various attack types (e.g., DDoS, Brute Force, Botnet).

The intrusion network data is read from the source into the Python environment using data analysis tool called pandas. The data is then loaded into a DataFrame with the help of the pd.read _csv() method for further processing. After the initial exploration of the data, a series of analysis is conducted on the data set. This includes exploring the nature of the target variable which is the label that represents the type of attack or Benign traffic. To determine the imbalance of any classes, bar plots and pie charts are used to represent the occurrence of each class in the data set.

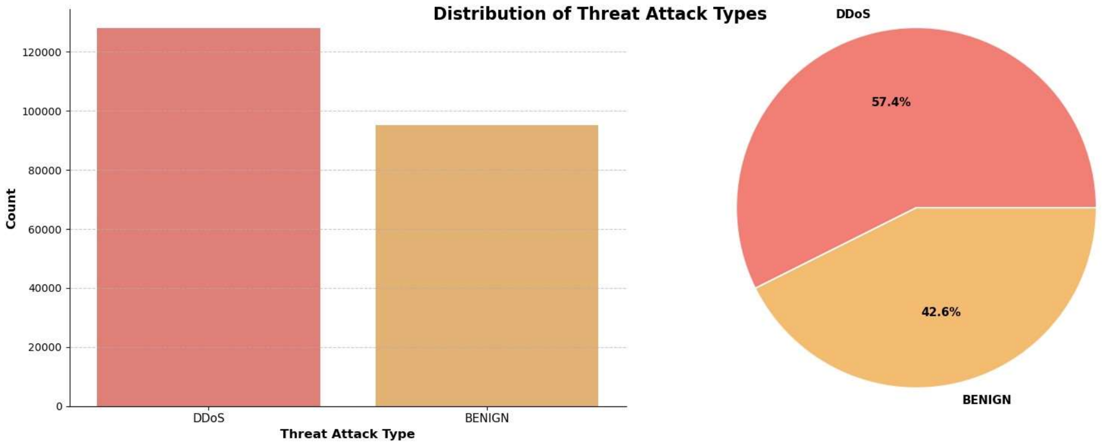


Figure 1: Distribution of Threat Attacks & Types

This is relevant for this discussion because an imbalance of data poses a risk of having skewed results from the built models and has to be handled if the case is present. Further, a correlation heatmap is generated to check the correlation of all numerical features where it demonstrate for any highly correlated variables that could be useful for model training operation.

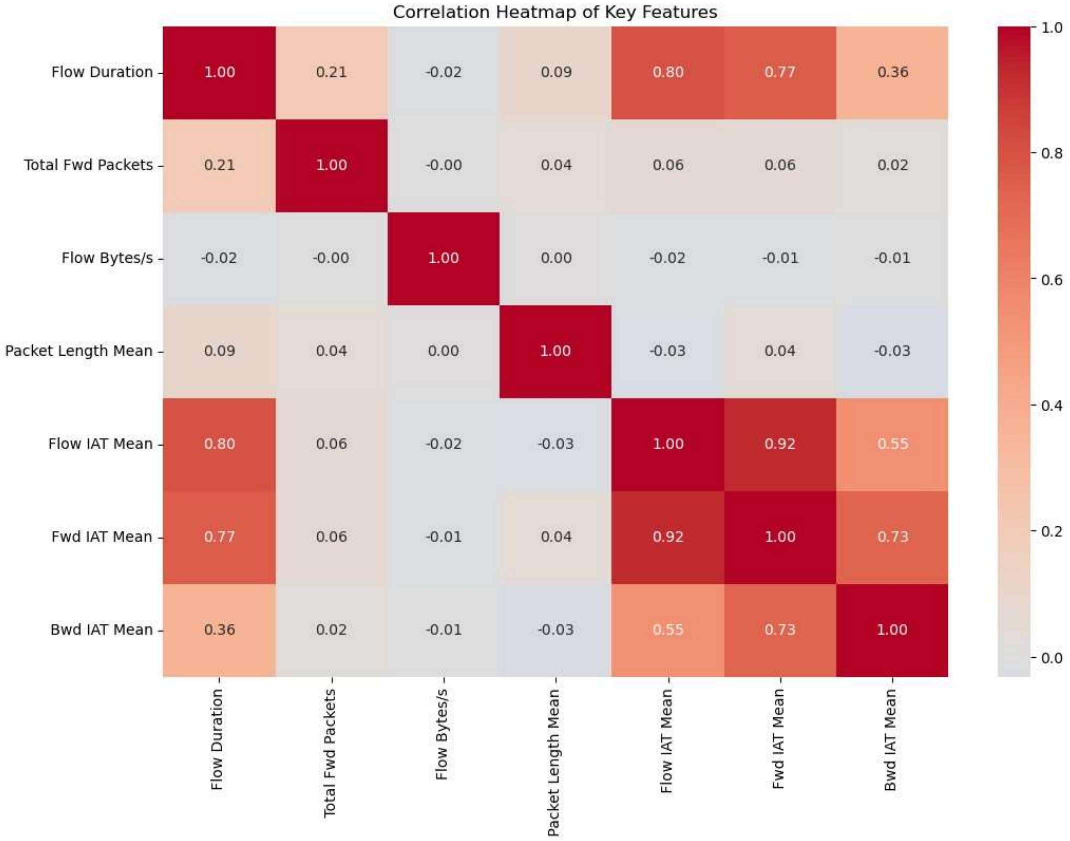


Figure 2: Correlation Heathmap for Network Traffic Dataset

3.3 Data Preprocessing & Feature Engineering

Data pre-processing follows the exploratory analysis in which the main aim is in preparation of data for modelling. To prevent noise from being introduced into the models, handling of missing values, handling of duplicate values, and handling of infinity in the dataset is performed. The missing values are imputed using fillna() method and infinity values are replaced by NaN and then imputed with a value of zero. The drop duplicates() method is used to remove duplicate rows also. As the data set has categorical variables these are converted into numerical form using Label Encoding so that the data should be more compatible with machine learning algorithms. Label encoding is used on each of the categorical column where the textual labels are replaced by values that are understandable by the models.

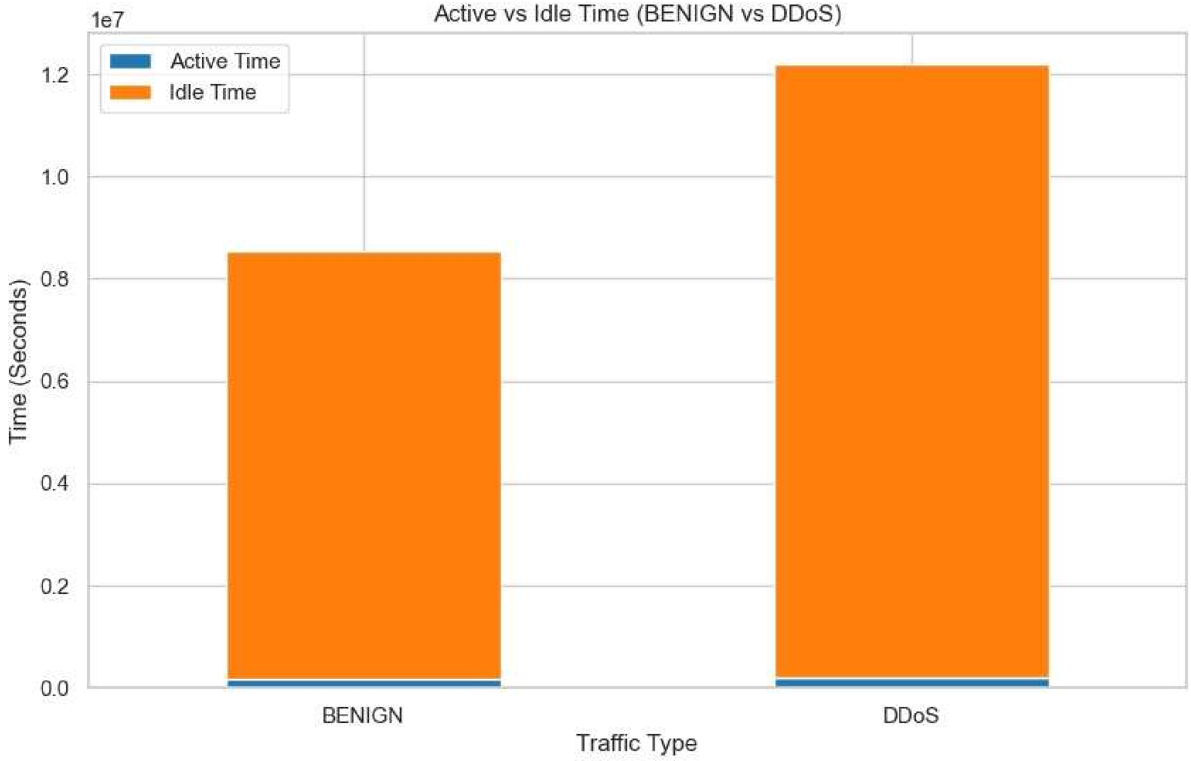
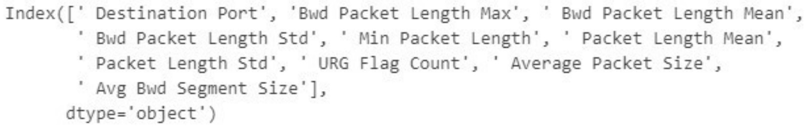


Figure 3: Proportion of ”active” and ”idle” times between BENIGN and DDoS traffic

The next methodology approach is featuring selection, the objective is to select the most important elements for further training. SelectKBest is used with the f classif scoring function, which determines the top 10 features as important. This makes the problem less complex by eliminating numerous variables that are not needed when training the models thus making the training faster. They are then used for subsequent analysis and further development of the model.



```
Index([' Destination Port', 'Bwd Packet Length Max', ' Bwd Packet Length Mean',
       ' Bwd Packet Length Std', ' Min Packet Length', ' Packet Length Mean',
       ' Packet Length Std', ' URG Flag Count', ' Average Packet Size',
       ' Avg Bwd Segment Size'],
      dtype='object')
```

Figure 4: Selected the Top 10 Features (Dimensional Reduction)

After pre-processing of the data, another step is to split the data set into the two sets where for training as well as testing data sets. This division makes sure that the models are trained from one part of the data and tested on the other part of the data hence giving an accurate estimate of the models performance. The data is divided in 80% for training and 20% for testing. This partitioning is done by the help of train test split() function from the Scikit-learn and the random state parameter is set in order to make the results reproducible.

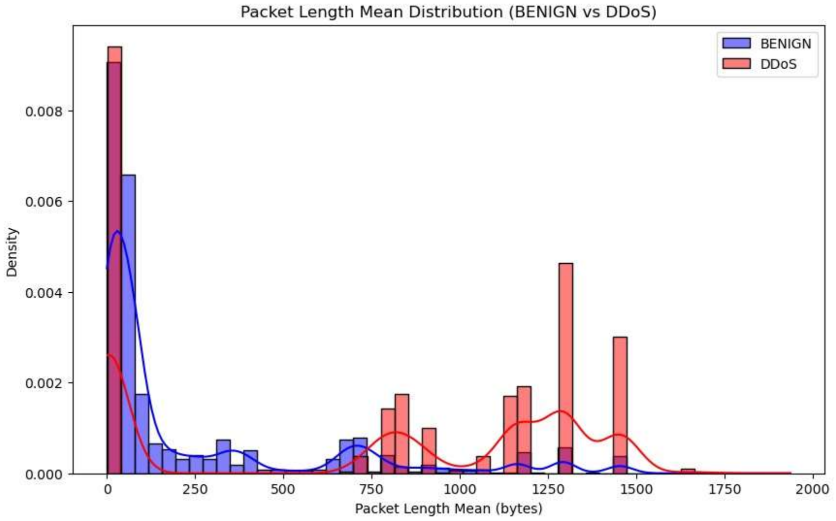


Figure 5: Packet Length Mean Distribution amoung BENIGN & DDos Attacks

3.4 Model Selection

Various machine learning models are chosen for the BENIGN and DDos attack traffic classification. The models used include:

1. K-Nearest Neighbors (KNN): KNN is an easy and non-parametric instance-based type of ML algorithm that allocates the data into classes based on the class of the closest neighbors. It doesn't train a model but it stores the dataset and computes distances for prediction. Despite this, KNN is computationally intensive, especially when applied on large datasets and has issues with high dimensionality data due to the curse of dimensionality.

2. Decision Tree: Decision Tree is a model which classify data into two or more subsets using features to form a tree structure. It's very explainable and can easily be represented graphically, but the problem is that if the tree is deep, it overfits. Some of these include pruning which can reduce this problem but some changes to the data can result in fluctuations in the model.

3. Logistic Regression: Logistic Regression is a simple and linear type of model that is employed for the binary classification and estimates the probability of an input in a given class by applying logistic function on the weighted sum of the features. Although effective with linearly separable data, it is not efficient with non-linear data and is also affected by outliers.

4. Random Forest: Random Forest is a technique of constructing many Decision Trees and using their results as an average to minimize the chances of over training. It can handle big data and offers feature importance and is less accurate but more costly than a single tree.

5. Deep Neural Network (DNN): The DNNs are structures of multiple layers of neurons that learn progressively advanced representations of data. They are useful for such tasks as image and speech recognition, the model can build complex nonlinear relationships. Still, they need a huge amount of data, are time-consuming, and are commonly considered 'black box' solutions because of their transparency.

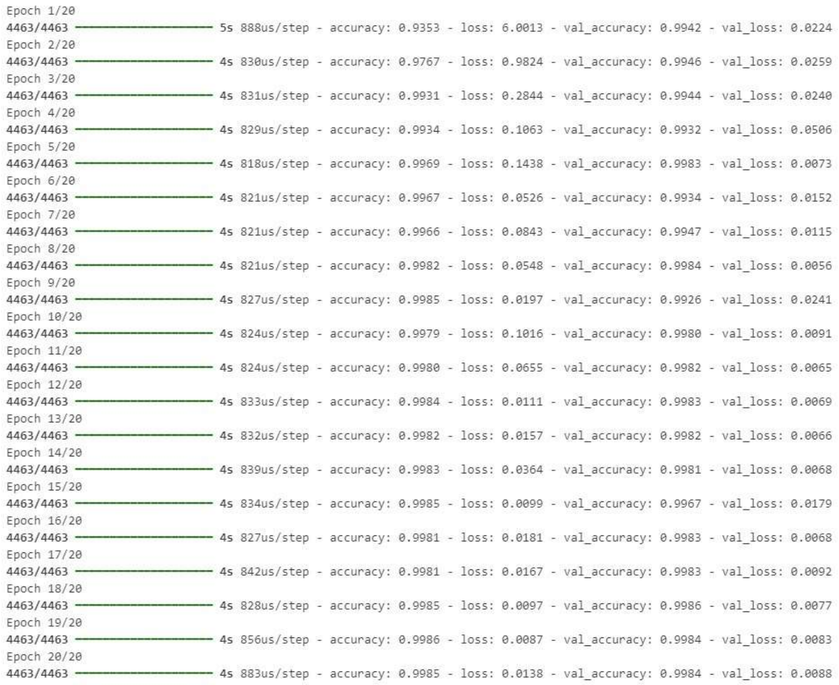
Table 2: Comparison between Models Strengths & Weaknesses

Model	**Strengths**	**Weaknesses**
K-Nearest Neighbors (KNN)	Simple, intuitive, no training phase	Computationally expensive for large datasets, sensitive to ”K”
Decision Tree	Easy to interpret, models nonlinear relationships, no scaling needed	Prone to overfitting, sensitive to noise
Logistic Regression	Simple, efficient, interpretable	Struggles with non-linear data, may underperform with complex features
Random Forest	High accuracy, reduces overfitting, handles large datasets well	Computationally expensive, less interpretable
Deep Neural Network (DNN)	Powerful for learning complex patterns, excellent for large datasets	Requires large data, computationally expensive, difficult to interpret

As each type of Machine Learning model has its advantages and disadvantages. Logistic Regression and K-Neighbors algorithms might show satisfying results for small and less complicated datasets and Random Forest & Deep Neural Networks for large and complex datasets respectively as experience in the case of Network Intrusion DataSet. The selection of the model is determined by the level of task, interpretability and computational cost.

3.5 Model Training and Evaluation

During the phase of training of the model, several ML algorithms are started and trained on the obtained data set. These are **Decision Tree Classifier**, **K-Nearest Neighbors (KNN)**, **Logistic Regression**, and **Random Forest Classifier** all of which are common and widely applied methods in machine learning for classification. Apart from these conventional models, another model that is used in this study to evaluate the performances of deep learning models is **Deep Neural Network (DNN)**. The deep neural network has a large number of layers and neurons that are aimed at identifying patterns that more shallow models cannot identify. Both models are trained with the training data set and then the prediction is made on the test data set. These predictions are then compared to the actual labels to check the accuracy of each model on the test set.



```
Epoch 1/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 5s 888us/step - accuracy: 0.9353 - loss: 6.0013 - val_accuracy: 0.9942 - val_loss: 0.0224
Epoch 2/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 830us/step - accuracy: 0.9767 - loss: 0.9824 - val_accuracy: 0.9946 - val_loss: 0.0259
Epoch 3/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 831us/step - accuracy: 0.9931 - loss: 0.2844 - val_accuracy: 0.9944 - val_loss: 0.0240
Epoch 4/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 829us/step - accuracy: 0.9934 - loss: 0.1063 - val_accuracy: 0.9932 - val_loss: 0.0506
Epoch 5/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 818us/step - accuracy: 0.9969 - loss: 0.1438 - val_accuracy: 0.9983 - val_loss: 0.0073
Epoch 6/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 821us/step - accuracy: 0.9967 - loss: 0.0526 - val_accuracy: 0.9934 - val_loss: 0.0152
Epoch 7/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 821us/step - accuracy: 0.9966 - loss: 0.0843 - val_accuracy: 0.9947 - val_loss: 0.0115
Epoch 8/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 821us/step - accuracy: 0.9982 - loss: 0.0548 - val_accuracy: 0.9984 - val_loss: 0.0056
Epoch 9/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 827us/step - accuracy: 0.9985 - loss: 0.0197 - val_accuracy: 0.9926 - val_loss: 0.0241
Epoch 10/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 824us/step - accuracy: 0.9979 - loss: 0.1016 - val_accuracy: 0.9980 - val_loss: 0.0091
Epoch 11/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 824us/step - accuracy: 0.9980 - loss: 0.0655 - val_accuracy: 0.9982 - val_loss: 0.0065
Epoch 12/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 833us/step - accuracy: 0.9984 - loss: 0.0111 - val_accuracy: 0.9983 - val_loss: 0.0069
Epoch 13/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 832us/step - accuracy: 0.9982 - loss: 0.0157 - val_accuracy: 0.9982 - val_loss: 0.0066
Epoch 14/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 839us/step - accuracy: 0.9983 - loss: 0.0364 - val_accuracy: 0.9981 - val_loss: 0.0068
Epoch 15/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 834us/step - accuracy: 0.9985 - loss: 0.0099 - val_accuracy: 0.9967 - val_loss: 0.0179
Epoch 16/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 827us/step - accuracy: 0.9981 - loss: 0.0181 - val_accuracy: 0.9983 - val_loss: 0.0068
Epoch 17/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 842us/step - accuracy: 0.9981 - loss: 0.0167 - val_accuracy: 0.9983 - val_loss: 0.0092
Epoch 18/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 828us/step - accuracy: 0.9985 - loss: 0.0097 - val_accuracy: 0.9986 - val_loss: 0.0077
Epoch 19/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 856us/step - accuracy: 0.9986 - loss: 0.0087 - val_accuracy: 0.9984 - val_loss: 0.0083
Epoch 20/20
4463/4463 ━━━━━━━━━━━━━━━━━━━━ 4s 883us/step - accuracy: 0.9985 - loss: 0.0138 - val_accuracy: 0.9984 - val_loss: 0.0088
```

Figure 6: Training of Deep Neural Network Model (DNN)

The following metrics are employed to evaluate the models in order to find that how effective they are in identifying cyber threats: The main measure of the general performance of each model is **Accuracy**, which calculates the ratio percentage of the correct predictions generated by the model. A **confusion matrix** is also applied to study true positive, true negative, false positive, and false negative values to get insights into the model's merits and demerits. Furthermore, the **classification report** is created to determine accuracy and make it possible to analyze the effective strength of the machine learning model in aspects of the recognition of different types of nettwork trafffic attacks in precision, recall and F1-score for each. Lastly, a comparison of the models is done with an evaluation of the performance metrics used for each and finally a bar chart to show the accuracy of each model. This comparison aids in determining the best model for the classification of network intrusion attacks and insight into the best and worst features of the different approaches to machine learning.

Hence, the methodology employed in this study is a rigorous approach to assessing AI-based threat detection solutions in cybersecurity. The research methodology of the study includes the exploration of data, pre-processing, model buidling & development, and model assessment to determine the best performing ML algorithm for the detection of DDoS attacks and benign type of traffic. Thus, the contribution of the research is in providing insights into the performance of different-different machine learning algorithms in identifying network intrusion that can improve cybersecurity measures.

4 Experimental Model Evaluation Results & Discussion

In this section, the evaluation results of the ML models employed for the task of classifying network intrusion attacks, specifically distinguishing between benign traffic and DDoS (Distributed Denial of Service) attacks. Five different models were tested, including Decision Tree, K-Nearest Neighbors (KNN), Logistic Regression, Random Forest, and Deep Neural

Network (DNN). Each of these models was evalauted based on its the classification accuracy, recall, precision, F1-score, and confusion matrix, all of which are important for understanding the effectiveness of the models in real-world intrusion detection tasks.

Table 3: Experimental Evaluation Results of Machine Learnnig Models

Model	**Accuracy**	**Precision (Benign)**	**Recall (Benign)**	**F1-Score (Benign)**	**Precision (DDoS)**	**Recall (DDoS)**	**F1Score (DDoS)**
K-Nearest Neighbors	96.94%	1.00	0.93	0.96	0.95	1.00	0.97
Decision Tree	96.19%	1.00	0.91	0.95	0.94	1.00	0.97
Logistic Regression	95.67%	0.99	0.90	0.95	0.93	1.00	0.96
Random Forest	98.26%	1.00	0.96	0.98	0.97	1.00	0.99
Deep Neural Network	**99.92%**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

4.1 K-Nearest Neighbors (KNN)

This K-Nearest Neighbors (KNN) model is evaluated with an accuracy of 96.94%, which is an indication of its efficiency in the classification of the benign and DDoS traffic. Further analysis of the classification report shows that the model has a precision of 1.00 for the benign traffic, this means that the model did not misclassify any benign traffic as the other type, meaning that the model did not produce any false positives. The DDoS class (label 1) had a precision of 0.95 which means that there were a few false positives. However, the recall for the DDoS class was 1.00, which means that all the DDoS attacks were detected. This reveals the high sensitivity of KNN in identifying the DDoS attacks as a strength of the algorithm. Precision and recall were also balanced with F1-scores of 0.97 for both classes, which is again a very good result. The confusion matrix for the KNN model revealed that it misclassified 1352 instances of benign traffic and only 13 instances of DDoS traffic. This implies that although KNN is effective, there could be some misclassification sometimes, particularly in identifying benign traffic. Although the KNN model is quite simple and efficient, its performance may decrease when the dataset is large or heterogeneous. Its reliance on the nearest neighbors for classification means that it is likely to be very slow if not optimized for the distance and number of neighbors on large data sets. However, KNN still proves to be a valuable model for situations where datasets can still be considered relatively small, providing high accuracy and recall for the detection of attacks.

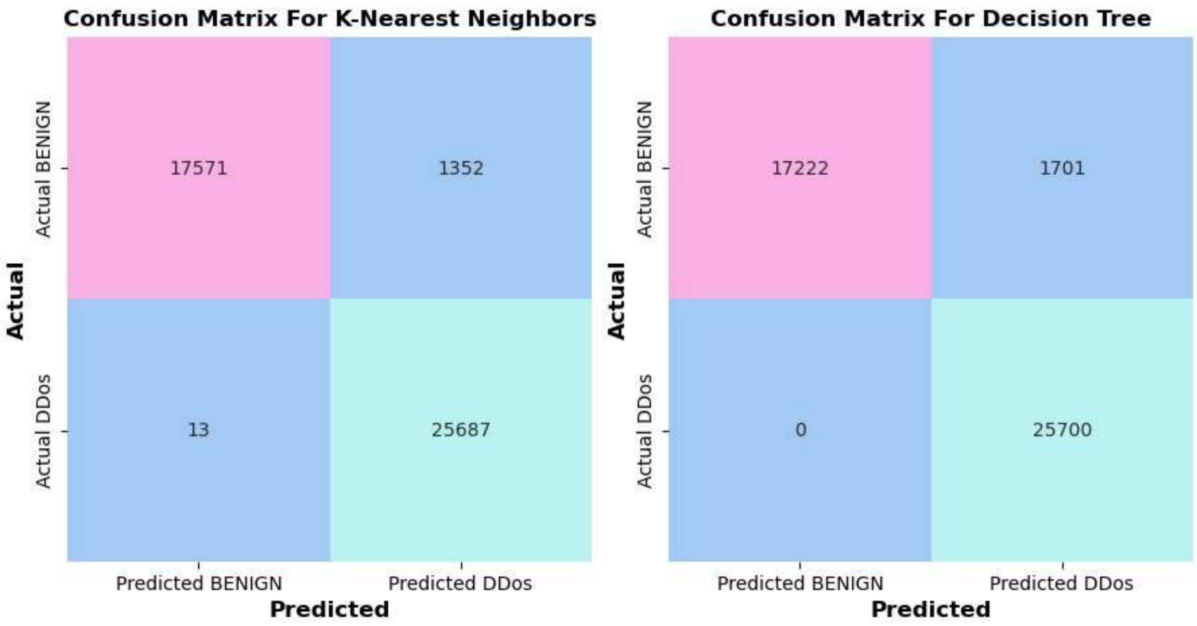


Figure 7: Confusion Matrix for K-Nearest Neighbors (KNN) & Decision Tree

4.2 Decision Tree

Decision Tree, the accuracy level was 96.19%, which is slightly worse than KNN but still rather impressive. When it comes to the accuracy of the model, it scored 1.00 in benign traffic, meaning it did not make any wrong call on benign traffic. While the recall for benign traffic was 0.91, meaning that the Decision Tree model incorrectly classified a great deal of benign traffic as DDoS. This is further evidenced by matrices where the confusion matrix alone reveals that 1701 benign instances were misclassified as DDoS. On the other hand, the Decision Tree was very successful in detecting DDoS attacks with a high precision of 0.94 and a recall of 1.00 meaning the model correctly classified all DDoS attacks but made a few false positive predictions on DDoS traffic. The precision for DDoS was 0.97 meaning the model can accurately identify attacks. The results of Decision Tree model show that it is very accurate in identifying DDoS attacks and the model might have problem in fine-tuning the decision boundaries between the benign traffic and attack traffic, where the Decision Tree model sometimes misclassifies the benign traffic. This is a common problem for Decision Trees because they tend to focus on certain patterns and noise in the data and, therefore, may have lower recall for the benign instances. However, even for simple or structured data, Decision Trees can be quite efficient and the interpretability of models is a considerable advantage of Decision Trees.

4.3 Logistic Regression

The Logistic Regression model resulted in accuracy of 95.67% which is slightly lower to KNN and Decision Tree models. In the classification report, the model presented a precision of 0.99 for benign traffic, so it rarely produced wrong positive predictions. Nonetheless, for benign traffic, the recall achieved was 0.90, which shows that 90 percent of benign traffic was misclassified as DDoS. This could be a concern in cases where traffic misclassification is harmless yet would prompt some action or alert. In case of DDoS traffic, the model yielded a precision of 0.93 and a recall of 1.00, which means that the model correctly identified all the DDoS attacks with no false negatives but had slightly higher number of false positives for the DDoS traffic compared to other models. The accuracy was 0.96 with F1- score for both classes, which shows that though the model's overall accuracy was slightly lower, the loss of accuracy has been balanced for both classes. From the confusion matrix it was observed that there were 1825 false negatives for benign traffic and comparatively few false positives for DDoS traffic. At the same time, being a quite simple and quite interpretable model, Logistic Regression had a poor recall rate of benign traffic, and one could suppose that in real-world scenarios the given model might require fine-tuning in order not to endanger legitimate traffic classification. This issue is typical of simpler models such as the Logistic Regression which could be less able to capture data patterns than the more complex models.

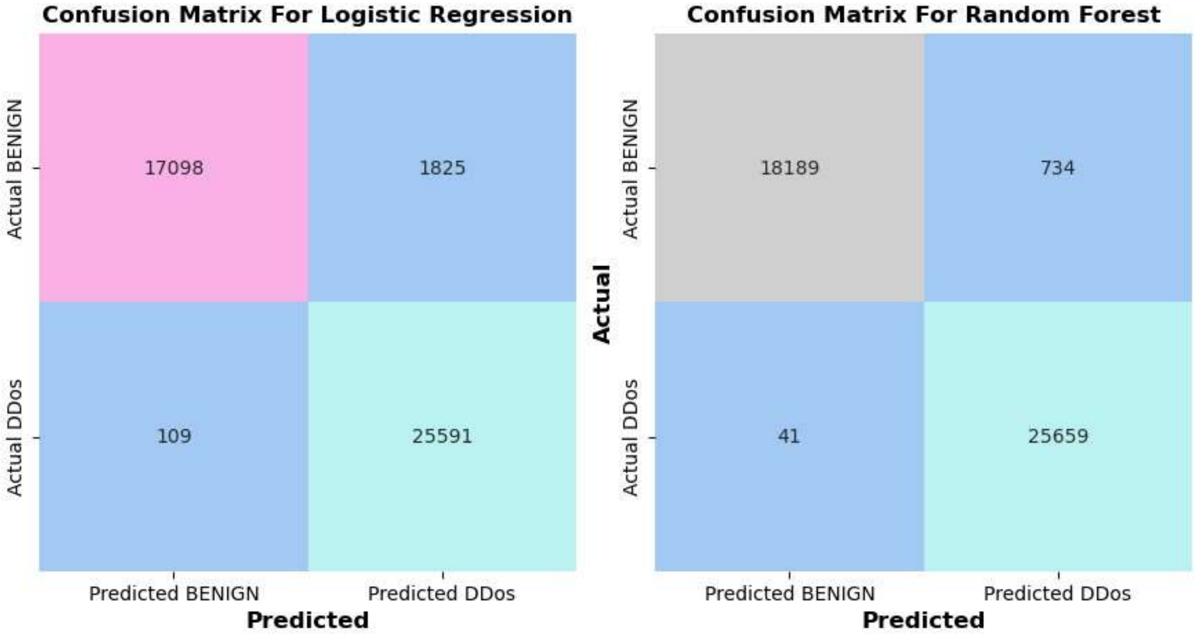


Figure 8: Confusion matrix of Logistic Regression & Random Forest

4.4 Random Forest

Random Forest model had the highest accuracy of 98.26% which was slightly lower than only the Deep Neural Network. With benign traffic, Random Forest had the best precision of 1.00 and recall of 0.96 which indicates that it was effective in the correct identification of benign traffic while at the same time having few false alarms. Specifically, for the DDoS traffic, the model achieved the precision of 0.97 and the recall of 1.00, which indicates that the proposed model did not generate any false negatives, i.e., all instances of DDoS attacks were detected. The precision/recall trade off was well balanced with F1-scores of 0.98 for both classes. The confusion matrix revealed that while misclassifying benign traffic, the Random Forest model had 734 misclassifications, and for the DDoS class, it had only 41 misclassifications, proving that the Random Forest model is very effective in correctly classifying both classes. Random Forest is an ensemble method which can work well with noisy data and non-linear correlations between variables, which is why it achieved such a high score in this test. Unlike a single

Decision Tree, Random Forest constructs multiple decision trees and then makes the final prediction by taking an average of the results thus minimizing on overfitting. This makes it especially useful in areas where precision is essential, and the data set extensive and intricate.

4.5 Deep Neural Network (DNN)

The Deep Neural Network model stands out with the highest accuracy of 99.92%, which is a significant improvement and a very good result compared to the rest of the models. On the training set, the DNN model had an accuracy of 99.91% while on the test set it had an accuracy of 99.92%. The results for the DNN model in the classification report were precision=1.00, recall=1.00, and F1-score = 1.00 for both benign and DDoS traffic. This implies that DNN model was able to classify all the benign and DDoS traffic correctly with no misclassifications. The confusion matrix for the DNN model revealed only 16 misclassified benign samples, and only 18 misclassified DDoS samples-which is quite an achievement, given the difficulty of the task.

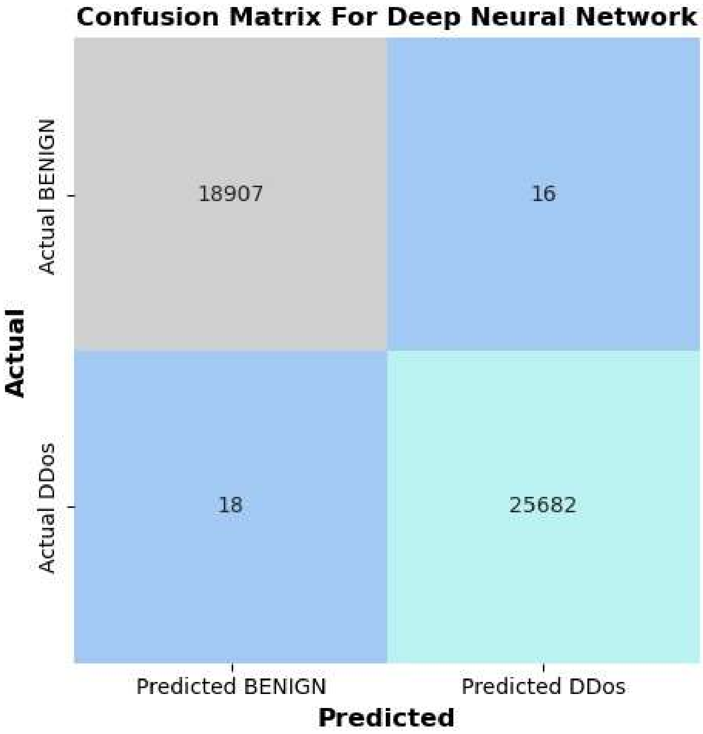


Figure 9: Confusion matrix of Deep Neural Network Model

The DNN performs very well due to its capacity to recognize the non-linear patterns of benign and malicious traffic in the data. The architecture structure of the neural network enables for the features of the input to be extracted at the lower layers of the model, and thus the model is highly effective for large-scale and complex data sets. Hence although the high accuracy and low loss values show that DNN is the best model for intrusion detection in this research, it should be noted that DNN’s are computationally intensive and may take longer to train as compared to the traditional models such as the Random Forest algorithm or Logistic Regression algorithm. As a result, the DNN might be more appropriate for the conditions, in which the accuracy is of the top importance and the resources are not a limiting factor.

4.6 Discussion

The findings represents that all the five models had good effective performance in aspects of accuracy, recall, precision, and generalization when assessed with the test set. The above three models; KNN, Decision Tree, and Logistic Regression achieved recognition accuracy of 95% to 97% while classifying benign and DDoS traffic with minor difficulties in classifying benign traffic. The Decision Tree model, in particular, had problems with the recall of benign traffic, whereas Logistic Regression had problems with false positives in DDoS detection.

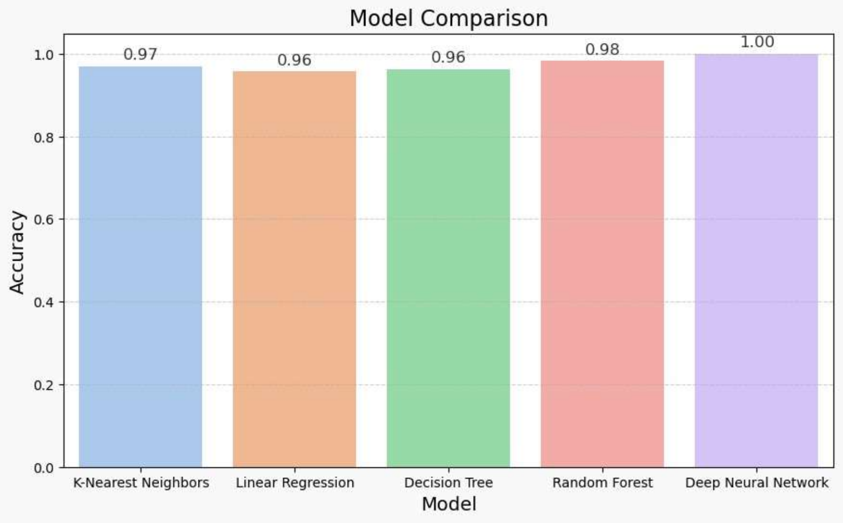


Figure 10: Comparison of Machine Learning Model Accuracy Results

The Random Forest model was shown to provide the highest accuracy and precision between the classes and the best interpretable model. Random Forest did not overfit, thanks to the decision not to aggregate the prediction that comes from each tree but rather taking the average of all trees, especially for more complicated data sets. Nevertheless, the DNN model presented the best accuracy of 99.92% with perfect value of the precision, recall and F1-score for both classes. The DNN outperformed the other models since it can learn complex & intricate patterns from the network traffic data, eliminating the requirement for the feature extraction.

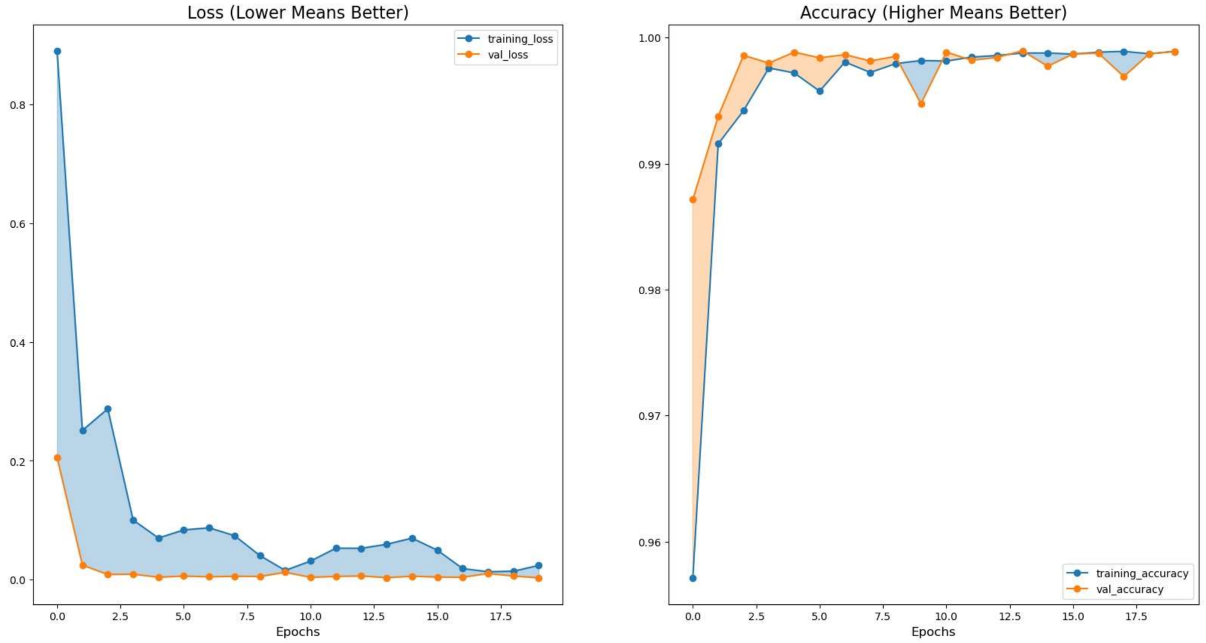


Figure 11: Deep Neural Network Model Training & Validation (Loss & Accuracy)

The DNN was the best-performing model but is also the most computationally intensive and resource hungry and so may not be immediately applicable to time-sensitive systems. Therefore, Random Forest may be more suitable for real-time applications where immediate results are required since it's a relatively accurate and quick model. However, deep learning models such as the deep neural networks are being incorporated in complex cybersecurity applications where detection rate is very important.

In summary, this research study demonstrates that the ML models, namely, RF and DNN, are the most efficient for traffic classification and DDoS attack identification. Conventional models like Logistic Regression & Decision Tree may still be relevant for simpler problems or when there are limitations on computing power and data size, however, the results of the DNNs indicate that they have the potential to radically transform intrusion detection systems when applied with big data and enough computing power. Further work may involve improving these models, looking at other methods of combining them, and adapting them for real time security applications.

5 Conclusion and Future Work

5.1 Conclusion

In this research, several classification models employing the approach of machine learning were used to recognize the traffic as either normal or DDoS attack. The models selected for this research study are the Decision Tree, K-Nearest Neighbors (KNN), Logistic Regression, Random Forest and Deep Neural Networks (DNN). Of all the models, the Deep Neural Network yielded the best results, with an accuracy of 99.92% and a perfect F1 score of 1 for both classes, Benign and DDoS. This means that the DNN model can clearly distinguish between the benign and DDoS traffic with very low probability of an error.

The Random Forest model also showed excellent performance in terms of accuracy at 98.26%, which indicates it is capable of addressing more intricate patterns in the data. The KNN, Decision Tree, and Logistic Regression models, though showing lesser accuracy, were equally useful in classification with a value ranging from 95.67%-96.94%. These results show that machine learning algorithms are very suitable for the purpose of DDoS detection in network traffic where real-time applications are possible.

However, while the DNN and Random Forest models showed the better results, all models have their own characteristics such as interpretability of Decision Trees and efficiency of KNN. Such variations imply that the choice of model could be determined by the nature of an application, for instance, if high interpretability or low time to inference in environments with limited resources are needed.

5.2 Future Work

Future work in this field could explore several avenues to further improve DDoS detection systems and expand their applicability:

- Real-time Detection: Although this research demonstrated strong performance in a controlled environment, real-world deployment of DDoS detection systems requires real-time analysis of incoming traffic. Future studies could focus on optimizing models for faster inference times, enabling the deployment of these models in network monitoring systems that can detect attacks as they happen.

- Multi-class Classification: This research focused on a binary classification task (benign vs. DDoS), but the world of cyber threats is more complex. Future work could involve expanding the model to handle multi-class classification, where different types of network attacks are classified, such as DoS, DDoS, SQL injection, and others.

- Cross-domain Evaluation: Finally, the models could be tested on different datasets to evaluate their generalizability. Since network traffic characteristics can vary significantly between different environments, validating these models on diverse datasets would help ensure their robustness across various network conditions and attack scenarios.

By pursuing these future directions, the field of DDoS detection can continue to evolve, creating more efficient, accurate, and interpretable models that can effectively safeguard networks against increasingly sophisticated cyber-attacks.

References

Agoramoorthy, M., Ali, A., Sujatha, D., TF, M. R. and Ramesh, G. (2023). An analysis of signature-based components in hybrid intrusion detection systems, *2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS)*, IEEE, pp. 1– 5.

Ahmad, W., Rasool, A., Javed, A. R., Baker, T. and Jalil, Z. (2021). Cyber security in iot-based cloud computing: A comprehensive survey, *Electronics* **11**(1): 16.

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): e4150.

Alhajjar, E., Maxwell, P. and Bastian, N. (2021). Adversarial machine learning in network intrusion detection systems, *Expert Systems with Applications* **186**: 115782.

Almomani, I., Ahmed, M. and Maglaras, L. (2021). Cybersecurity maturity assessment framework for higher education institutions in saudi arabia, *PeerJ Computer Science* **7**: e703.

Cadet, E., Osundare, O. S., Ekpobimi, H. O., Samira, Z. and Wondaferew, Y. (2024). Ai-powered threat detection in surveillance systems: A real-time data processing framework.

El-Hasnony, I. M., Barakat, S. I., Elhoseny, M. and Mostafa, R. R. (2020). Improved feature selection model for big data analytics, *IEEE Access* **8**: 66989–67004.

El Morr, C., Jammal, M., Ali-Hassan, H. and El-Hallak, W. (2022). Data preprocessing, *Machine Learning for Practical Decision Making: A Multidisciplinary Perspective with Applications from Healthcare, Engineering and Business Analytics*, Springer International Publishing, Cham, pp. 117–163.

Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*, MIT Press.

Hasib, K. M., Showrov, M. I. H., Al Mahmud, J. and Mithu, K. (2022). Imbalanced data classification using hybrid under-sampling with cost-sensitive learning method, *Edge Analytics: Select Proceedings of 26th International Conference—ADCOM 2020*, Springer Singapore, Singapore, pp. 423–435.

Kavitha, D. and Thejas, S. (2024). Ai enabled threat detection: Leveraging artificial intelligence for advanced security and cyber threat mitigation, *IEEE Access* .

Khando, K., Gao, S., Islam, S. M. and Salman, A. (2021). Enhancing employees information security awareness in private and public organisations: A systematic literature review, *Computers & Security* **106**: 102267.

Kothamali, P. R. and Banik, S. (2022). Limitations of signature-based threat detection, *Revista de Inteligencia Artificial en Medicina* **13**(1): 381–391.

Mallick, M. A. I. and Nath, R. (2024). Navigating the cyber security landscape: A comprehensive review of cyber-attacks, emerging trends, and recent developments, *World Scientific News* **190**(1): 1–69.

Mallikharjuna Rao, K., Saikrishna, G. and Supriya, K. (2023). Data preprocessing techniques: emergence and selection towards machine learning models-a practical review using hpa dataset, *Multimedia Tools and Applications* **82**(24): 37177–37196.

Mensah, K. and Boateng, A. (2024). Adversarial machine learning: Understanding and mitigating vulnerabilities, *Advances in Computer Sciences* **7**(1): 1–9.

Mohamed, N., Taherdoost, H. and Madanchian, M. (2024). Review on machine learning for zero-day exploit detection and response, *International Conference on Smart Technology*, Springer Nature Switzerland, pp. 163–176.

Mohan, P. V., Dixit, S., Gyaneshwar, A., Chadha, U., Srinivasan, K. and Seo, J. T. (2022). Leveraging computational intelligence techniques for defensive deception: a review, recent advances, open problems and future directions, *Sensors* **22**(6): 2194.

Pavithra, S. and Vikas, K. V. (2024). Detecting unbalanced network traffic intrusions with deep learning, *IEEE Access* .

Pureti, N. (2022). Zero-day exploits: Understanding the most dangerous cyber threats, *International Journal of Advanced Engineering Technologies and Innovations* **1**(2): 70– 97.

Rahaman, M., Pappachan, P., Orozco, S. M., Bansal, S. and Arya, V. (2024). Ai safety and security, *Challenges in Large Language Model Development and AI Ethics*, IGI Global, pp. 354–383.

Sarker, I. H., Furhad, M. H. and Nowrozy, R. (2021). Ai-driven cybersecurity: an overview, security intelligence modeling and research directions, *SN Computer Science* **2**(3): 173.

Sharma, P. and Sharma, R. (n.d.). A review on machine learning applications in cyber security.

Skopik, F., Wurzenberger, M., Ho¨ld, G., Landauer, M. and Kuhn, W. (2022). Behaviorbased anomaly detection in log data of physical access control systems, *IEEE Transactions on Dependable and Secure Computing* **20**(4): 3158–3175.

Uchendu, B., Nurse, J. R., Bada, M. and Furnell, S. (2021). Developing a cyber security culture: Current practices and future needs, *Computers & Security* **109**: 102387.

Wiafe, I., Koranteng, F. N., Obeng, E. N., Assyne, N., Wiafe, A. and Gulliver, S. R. (2020). Artificial intelligence for cybersecurity: a systematic mapping of literature, *IEEE Access* **8**: 146598–146612.