Optimizing Real-Time DDoS Detection with Autoencoders for Enhanced Cybersecurity

MSc Research Project
MSc in Cybersecurity

Raj Bharath Murali
Student ID: X22240888

School of Computing
National College of Ireland

Supervisor: Michael Pantridge

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Raj Bharath Murali
Student ID:	X22240888
Programme:	MSc in Cybersecurity
Year:	2024
Module:	MSc Research Project
Supervisor:	Michael Pantridge
Submission Due Date:	12/08/2024
Project Title:	Optimizing Real-Time DDoS Detection with Autoencoders for Enhanced Cybersecurity
Word Count:	8313
Page Count:	26

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:	Raj Bharath Murali
Date:	11th August 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):	

Optimizing Real-Time DDoS Detection with Autoencoders for Enhanced Cybersecurity

Raj Bharath Murali
X22240888

Abstract

The emerging threats of Distributed Denial-of-Service (DDoS) attacks and issues with previous models have raised concerns about the security of cyberspace. If critical systems are compromised, the results could range from failure to life-threatening situations. Previous strategies have primarily been designed for traditional machine learning, and rule-based methods are not as effective for identifying new attacks. This research utilized deep learning algorithms with an emphasis on autoencoders to solve the DDoS detection challenge. Through our analysis, we discovered that autoencoders are more accurate and have higher precision, recall, and F1-score than the CNN and RNN models we tested. The principal innovation of our work was creating a real-time web application that incorporates the autoencoder model, which can be used in a live environment to automatically detect and defend against DDoS incidents. We feel that our contribution is unique and that it will be welcomed by the community, as it offers a scalable approach that can be applied to a variety of fields and can be used to test future detection strategies.

1 Introduction

Technological advances have emerged as a promising solution to the shifting global environment where individuals and businesses are both benefitted. However, amid this promise, modern technology has presented an immense degree of cyber threat to the network and system. The significance of managing cybersecurity in the contemporary ecosystem is undeniable. The prominence built through digital communication, mobile computing, cloud computing, Internet of Things (IoT) has not only surged the advances in business performance and individuals' living standards but also increased cyber-attacks. Evidence in the literature has marked the importance of cybersecurity for the protection of computer systems as well as networks against various unauthorized intrusions, malicious disruptions in the system, data theft, and damage and interruption of services. Salih and Abdulrazaq (2024) explained that cyberspace has been extremely dynamic where new attacks are becoming highly sophisticated with its development by hackers. Among these attacks, the "distributed denial of service" (DDoS) is a renowned cyber-attack that purposively exhausts the targeted network system with malicious traffic. Indeed, different statistical methods have been designed and introduced to detect DDoS attacks; however, developing a real-time detection method with a "low computational overhead" has been an underdeveloped area. Accordingly, evaluating a new detection model depends on the existence of a well-designed dataset(s).

The existence of DDoS attacks as a potential cyber security threat has gained significant attention in research. This is because there has been an undeniable increase in DDoS attacks in recent times. According to the report presented by Kapko (2024), the elevation of malicious DDoS attacks has coincided with the mass exploitation of the zero-day vulnerability, “HTTP/2 Rapid Reset” that threatened actors that were focused on launching DDoS attacks last year. Amid this fact, there has been an extensive increase in DDoS attacks recorded in 2023, which is more than the estimation recorded in the previous two years. As per the statistical report presented by Yoachimik and Pacheco (2024), in 2023, approximately 5.2 million “HTTP DDoS attacks” consist of more than 26 trillion requests recorded and mitigated through automated defense mechanisms. Understanding the record, it has undeniably provided new insights into the size as well as the sophistication of the malicious attack with the wider internet community such as Cloudflare. That has faced a persistent dilemma with the consistently engineered campaign comprising several hyper-volumetric (HTTP) DDoS attacks that were minimally existent in previous times. As a part of the campaign, during the third quarter (Q3), the system has mitigated one of the biggest attacks with more than 201 million requests detected per second which is nearly 8 times higher than the previous year’s (2022) record, which was 26 million requests.

Focusing on ways DDoS attacks can be specified, it has been identified that over the years, the intrusion has become more and more sophisticated with fewer capabilities, the requirement of resources & time, thus ginning significant attention from the cybersecurity department. At present, industries are growing extensively with the integration of modern technologies. A significant level of dependence on cloud and IoT infrastructure has further amplified the risk of malicious attacks. As reported, the prime targets of DDoS attacks are the retail segment, shipment and logistics, and tourism and hospitality sectors Kapko (2024). Upon understanding the concern, a prominent approach to efficient detection of DDoS has become mandatory and relevant to which distinct focus has been given to different detection methods and classification models. At the same time, priority has been given to dataset selection, which is a fundamental aspect of the effective detection of cyber threats. The consideration of a reliable model depends on how it is trained and the effectiveness of the dataset. Salih and Abdulrazaq (2024) explained that the validation of the robustness, as well as the effectiveness of the model, can be identified by testing the same with suitable datasets. Understandably, the CIC-DDoS2019 dataset has gained specific attention in recent times, which is a class imbalance dataset with a large content containing various DDoS attacks.

According to the information presented by UNB (2019), the CIC-DDoS2019 dataset contains benign and updated DDoS attacks that resemble the exact real-world data (eg. PCAPs). The dataset includes certain results of “network traffic analysis” where the CICFlowMeter-V3 is used to label the flow depending on the time-stamp, sources & destinations IPs, and sources & destination parts as well as protocols. Upon focusing on the significance of this dataset, generating “realistic traffic” in the background is an utmost priority for its effective development. For a suitable choice, developers typically utilize the “abstract behavior” from 25 users of HTTP, HTTP, and FTP links as well as email protocols. The distinct purpose served by this dataset has indeed gained popularity in recent times due to the features it possesses and the outcomes that can be established. While understanding the importance of the dataset, the information outlet has further laid a focus on the methods and advanced detection models that have been a growing trend. Among all the techniques that have been established for years, an advancement toward deep learning models has received significant attention because of its relevance based on interpretability, speedy outcome, resource efficiency, and satisfactory performance. The priority given to these deep learning models is a precise configuration of advanced machine learning (ML) algorithms although some models exist as neural architecture that provide improved experimental results compared to existing methods when trained and tested with enhanced datasets.

The importance that cybersecurity has received in the current high-tech environment has served as a basis for applying improved detection models in practice for an effective result, especially with DDoS attacks. In the modern environment, novel DDoS attacks have been exhibited with notable complexities, which are typically characterized by dynamicity from multiple vectors followed by continuous evolution and rapid expansion. Some of these vectors that are identified in studies include “volumetric attacks” containing a certain bandwidth consumption, “TCP state exhaustion attacks”, and “low-and-slow application layer attacks”. In identifying the proximal threat related to these DDoS attacks, modern detection methods are highly imperative to consider to accomplish a potential detection accuracy. Thus, the current study aimed to investigate the effectiveness of deep learning models in detecting DDoS attacks using the CIC-DDoS2019 dataset.

1.1 Research Objectives

We will try to achieve the following research objectives to enhance the detection of DDoS attacks using advanced deep learning methods in real-time toward better cybersecurity.

- To develop a robust deep learning-based system to detect and prevent DDoS attacks in real-time, securing network infrastructures.
- To evaluate Deep Learning Models for Optimal Security and Compare the performance of CNN, RNN, and Autoencoders for DDoS detection, focusing on accuracy, precision, recall, and F1-score.
- To identify and deploy the most effective model, particularly Autoencoders, in a web application to secure systems against DDoS attacks.
- To create a client-server web application that detects DDoS attacks and applies appropriate mitigation strategies in real-time.

1.2 Research Question

- How can deep learning, particularly Autoencoders, be effectively utilized to enhance real-time detection and mitigation of DDoS attacks in a practical, deployable cybersecurity framework?

2 Related Work

The current chapter has emphasized the literature evidence that has been extensively explored to determine the detection process of DDoS attacks. As outlined in the previous chapter academic researchers and developers have traced the pattern of recognizing DDoS attacks through developing and stating the relevance of different techniques. This chapter, therefore, assessed the information from existing studies based on which cybersecurity issues are distinctly explored. In this discussion, priority has been given to various datasets that are used to train and test models to enhance the detection process. At the same, a contrast and comparison have been established on these methods to identify the reliability and accuracy of the technique in DDoS attack detection.

2.1 DDoS Attack Detection Using Machine Learning Algorithms

The “Distributed Denial of Attacks” (DDoS) has received wide research interest in recent times because of the increasing cyber threat acknowledged on the network system. With the extensive application as well as the evolution of the cyber world, network attacks have become a real concern Aktar and Yasin Nur (2023). Information presented by Abood and Abdul-Majeed (2024) has explained that persistent network attacks are affecting users’ systems by consuming resources based on spontaneous requests rather than acknowledging legitimate requests. Amid this concern, rigorous attention has been given to various methodologies that exist to detect and mitigate such network attacks. Amitha and Srivenkatesh (2023) explained that the application of “Radial Basis Function” (RBF), which is a class of “Artificial Neural Networks” (ANN), is a commonly used method that enables functions approximation, recognition of patterns, and tasks classifications. Understandably, the above study introduces a hybrid framework of the RBF-LSTM model which is a neural architecture configuration that has been enhanced to detect DDoS attacks by improving the cloud security and computing infrastructure. The study has utilized features through data preprocessing from the CIC-DDoS2019 dataset based on which the effectiveness of the model is identified. On the contrary, another study presented by Fadlil et al. (2017) has explained the frequency of occurrence of a network attack and the impact on users’ systems.

Notably, the study has introduced the significance of a statistical Naive Bayes (NB) method that has gained popularity because of its efficient network traffic detection using statistical analysis. Fadlil et al. (2017) further explain that the statistical approach to detecting DDoS attacks has shown a relationship with the “Intrusion Detection System” (IDS) that can predict existing attacks. Wakamiya et al. (2024) on the other hand, explained that a Random Forest (RF) classifier, which is a machine learning model, has proven effective with its accuracy rate of 99.97% in the detection of complex DDoS attacks using the CIC-DDoS2019 dataset. The identification of “Denial of Service” (DoS) attacks has become a challenging task with the increasing sophistication of malicious software and their invasion of the network system. According to the explanation provided by Kumari and Mrunalini (2022), it is indeed understandable that many improvements in the detection of DDoS attacks have been enhanced. However, the inhibition capacity against the server’s ability to provision resources to genuine customers has set certain challenges in resource use.

Date-Author	Key Findings/Results	Advantages	Limitations
Bravo and Mauricio (2019)	Findings show that modern methods exploit specific characteristics of the attack including traffic patterns, users' requests, and certain tools to identify the attack type and retain the highest accuracy.	Introduces a systematic review that discloses information on DDoS attacks and acknowledges the importance of introducing an improved detection method.	Information based on systematic review without a focus on any particular model
Li et al. (2023)	This study provides comprehensive insights into DDoS attacks and vulnerabilities that have emerged with the novel attacks. To overcome this, a "SOTA defense solution" has been introduced depending on programmable switches.	Provide extensive insights into the effectiveness of the defense system, due to its deployment flexibility. Also, detection capabilities and robustness against the adversarial attack are unmatchable.	Data gathered based on a survey without proper focus on detection models, thus highlighting the need for an experimental study in the future to examine models' detection accuracy and adoption flexibility.
Bahashwan et al. (2023)	Findings established from the review show that the extant literature-specific knowledge is scarce and needs extensive research to determine the open issues with more clarity.	The overall review has presented a corresponding significance of hybrid ML-DL approaches in the detection of existing and zero-day attacks using "private synthetic datasets".	Information based on review, that concerns data validity and model reliance for effective detection of the network attacks.
Najar and Manohar Naik (2024)	Findings show a reliable outcome with the model that effectively combats DDoS attacks and also safeguards the seamless activity of network operations.	Introduces the importance of SDN which has gained popularity in contemporary times due to its agility and flexibility regarding network management.	No specific information on the dataset used for model training and testing.
Mustapha et al. (2023)	As per findings GAN is highly efficient in mimicking legitimate data. Experimental result shows that the LSTM detection method is highly efficient in identifying GAN-specific DDoS traffic with an accuracy rate varying from 91.75-100%.	Provided insights into the challenging cyber-security issues while addressing the concern by introducing different ML/DL techniques.	Uncertainties with evasion of ML/DL techniques for developing attack traffic.

Table 1: Summary of Literature Review

Under such a situation, the evidence presented by Kumari and Mrunalini (2022) identifies the relevance of machine learning models such as logistic regression, na¨ıve Bayes and random forest in DDoS attack detection. Using the CAIDA-2007 dataset, it has been identified that each model upon training with these datasets provides improved results than existing methods. Contrastingly, in another study presented by Saini et al. (2020) stated that DDoS attacks remain a challenge with the continuation of false requests to legitimate users rather than facilitating genuine services. In the above study, the detection of malicious network traffic has been enhanced through a model validation using a dataset containing data on HTTP flood, SID-DDoS and further normal traffic. Experimental results obtained from the study indicated that the ML tool used, WEKA has successively classified different attacks while the J48 algorithm produces better detection results than existing other classifiers such as random forest (RF) and Na¨ıve Bayes (NB).

2.2 DDoS Attack Detection Using Deep Learning

Algorithms The detection of DDoS attacks to improve the convenience of network security has become a relevant measure acknowledged by research experts and developers. In this regard, Behal et al. (2017) explained that it is a challenging aspect that DDoS attacks nowadays are using the "HTTP protocols' logical semantics" to intermix malicious requests with legitimate requests. Evidence presented in another study by Kumar et al. (2023) has introduced an early detection method using a deep neural architecture - Long-Sort-Term-Memory (LSTM) using features from the preprocessed data of the CIC-DDoS2019 dataset. The study precisely discloses the information on the model's accuracy where the accuracy rate achieved is 98%. Understanding the implications of the accuracy rate, it is suggestive to explain that the neural network has outperformed many existing ML models that have been trained with a similar dataset to detect novel DDoS attacks. Hadi (2024) explained that DDoS attacks are a common threat to cloud computing infrastructure, especially when complexities and sophisticacy in the attack pattern are continuously identified. Understandably, Hadi (2024) presented a hybrid model based on an autoencoder and CNN framework that have been trained using the NSL-KDD dataset. The experimental result obtained from the study shows that the model has achieved an accuracy level of 97.7% followed by reliable performance with other parametric values.

Information presented by another study, Najar and Manohar Naik (2024) explained that a recent research approach has presented the importance of the CNN model - a deep neural architecture, which shows its potential in DDoS attack detection. Mustapha et al. (2023) further explained that an ensemble approach of "Balanced random Sampling" (BRS) and "Convolutional Neural Network" (CNN) has presented an accuracy level of 99.99% in the detection rate, which compared to existing methods is significantly higher. The advancement in the network infrastructure ha posed significant threats to humans apart from facilitating improved online services. The typical condition of DDoS attacks has become a prevalent concern that needs prominent solutions to reduce cyber risks. In this regard, the information presented by Tekleselassie (2021) explained that the utilization of a successive model that combines deep infrastructure with a knowledge-graph classifier has served a promising solution. It has been observed that the model is highly flexible in application and easily expandable to wide areas of the detection process. While trained and tested with the CIC-IDS-2017 dataset containing approximately 53,127 events, the model has provided an accuracy of 99.97%, thus indicating its significance in the detection process.

2.3 DDoS Attack Detection Using Hybrid Methods

DDoS attacks typically occur when cyber attackers target users' systems or networks' resources and make them unavailable by flooding malicious requests. Generally, attackers exploit network and system vulnerabilities across different protocols (eg. HTTP, ICMP, and others), transport systems, and network infrastructure for sending malicious payloads. The proper addressing of the concern therefore has been enhanced by introducing a hybrid model - "Autoencoder-Multiple-Layer Perceptron Network" (AE-MLP) Wei et al. (2021). The model uses reduced or compressed features from the CIC-DDoS2019 dataset and demonstrates an accuracy level of 98% with her F1-score. In another study presented by Bahashwan et al. (2023) presented the contribution of a hybrid approach that overlaps the pattern recognition and classification features of both ML and DL models. Findings presented by the above study further inform that "private synthetic datasets" and "unrealistic datasets" are common datasets used by experts for effective evaluation and specification. In another study presented by Bravo and Mauricio (2019), the author explained that the detection, as well as defense mechanism, has received greater importance in the detection of DDoS attacks. In this regard, the above study has demonstrated the significance of specific network traffic characteristics, the implication of users' responses, and the importance of datasets as well as models. Upon presenting the information, it can be stated that the detection models have presented spectacular importance based on their significant contribution to designing an appropriate strategy to neutralize the attack.

Notably, Li et al. (2023) explained that the upscale of the DDoS attack is mainly identified in IoT devices and cloud networks due to which disruptions are a common concern in individual and business operations. Apart from this, it is also evident that collaboration specifically between certain domains and "inter-domain resource scheduling" has become a prominent issue when designing a cooperative defense mechanism. Therefore, research has specifically drawn attention to the introduction of a "SOTA defense solution" which contains programmable networks with programmable switches. Despite this fact, more focus is needed to establish a comprehensive defense mechanism that can ensure proper network security. Adversarial network attacks are a standard problem that has efficiently mimicked legitimate data. Particularly, the "Generative Adversarial Networks" (GAN) is highly effective in perceiving such network attacks. Upon understanding the concern, the evidence presented by Mustapha et al. (2023) has explained the contribution of different ML/DL models in DDoS detection. As per the experimental result confirmed, the LSTM-RNN model has presented an outstanding accuracy between 91.75 and 100% with different datasets.

2.4 Literature Gap

Current literature on DDoS detection has largely disregarded Autoencoders in favor of traditional machine learning and deep learning models, as well as not often addressing practical implementation scenarios for designing a deployable web application that functions in real-time. This gap connects the advancements made in theoretical device learning with practical applications in the real world using the CIC-DDoS 2019 dataset.

Our study addresses this issue by applying Autoencoders for DDoS detection and incorporating a deployable web application with the use of Autoencoders. This is a new point to view of DDoS detection and mitigation in real time, and is a meaningful contribution to cybersecurity.

3 Methodology

In the technological world, the use of smart devices and services over the internet has become rampant, hence the issue of cybersecurity becomes essential. If talk about the threats businesses and individuals are confronted with, DDoS (Distributed Denial of Service) attack is the most common type of attack. It is a cybercrime activity in which several systems conspire to overwhelm thereby making it truly unusable for the resultant effect. The fundamental architecture attack of the internet is exploited by the DDoS attack that leverages the number of devices to send an overwhelming number of requests to a victim's system. These attacks can cause financial loss and reputational damage. There are several strategies have been mitigated to prevent these DDoS attacks. The strategies are technical measures, such as traffic filtering, rate limiting, and using CDNs (Content Delivery Networks). In order to accurately predict the different types of DDoS attacks, we have designed an methodology, which consists of a certain set of steps. Each step plays and crucial role in achieving our research objectives. After preprocessing the data, deep learning algorithms are applied to the data. Here is the detailed description of each section with an architectural diagram shown in Figure 1.

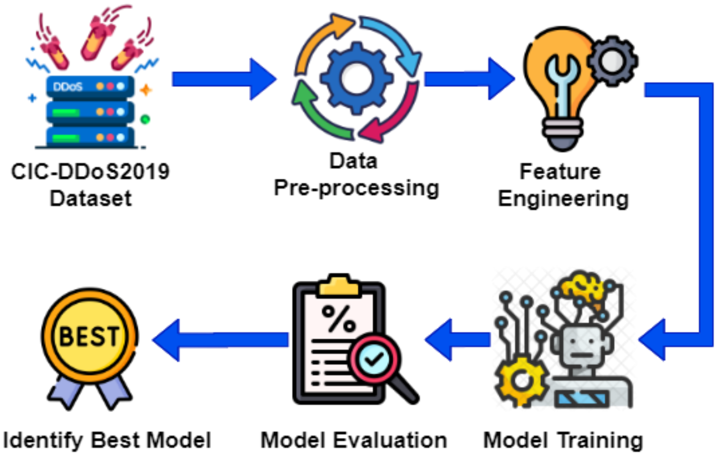


Figure 1: Methodology diagram for Classifying DDoS attacks

3.1 Data Description

In this research, the dataset is taken from the Canadian Institute for Cybersecurity which is a newly generated dataset CICDDoS2019. The dataset CICDDoS2019 consists of benign and different types of DDoS attacksUNB (2019). The dataset contains the result of the traffic analysis of the network using CICFlowMeter-V3 having labels based on timestamp, source and destination IPs, destination and source ports, protocols, and attack. The dataset contains a total of 409000 rows and 67 attributes which will demand high resources for analytics, model building, model compiling, and training. Also, the dataset contains different DDoS attacks such as PortMap, NetBIOS, LDAP, MSSQL, etc. The target column ‘label’ has 13 distinct values, which is a high number of classes for classification, thus we need to reduce the number of classes which can be done by mapping these classes into some major classes defined in the documentation of this data. Each class of DDoS attacks is shown in Figure 2 .

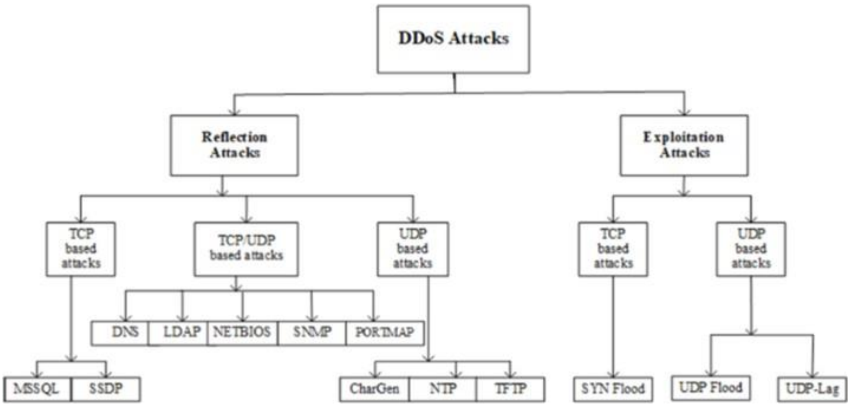


Figure 2: Different Classes of DDoS attacks

3.2 Data Preprocessing

Data preprocessing is an important part of data analysis and model building. It consists of the preparation of raw data for analyzing and transforming the data into a clean, organized, and suitable format. Data preprocessing is important because the raw data contains noise, missing values, and duplicate values that can poorly impact the performance and the accuracy of the models. Data preprocessing improves the quality of the data by cleaning and organizing the data; by preprocessing the data the performance of the model can be improved by making patterns in the data and reducing noise. In this study, unnecessary columns are removed and the duplicate values are dropped from the data. The classes are mapped with the actual class and the WebDDoS class is dropped since there is no mapping for this class. After mapping the classes, we have identified imbalanced data, where we identified that Benign classes have very few rows while the total records are very large in number, therefore a sample of 100,000 records is taken so that all benign rows get included to preserve them while sampling. After preprocessing, data sampling has been applied to the data, where we have considered the data with 107213 rows and 66 columns.

3.3 Exploratory Data Analysis

Exploratory Data analysis (EDA) is the process of assessing and characterizing the data which is done through examining, purging, structuring, and finally modeling the data and which is useful in finding the relationship between the data, making conclusions, and decision-making. It is the main part of the ML process that makes it possible to derive important data from a mass of data. Here are some analyses carried out by analyzing the data.

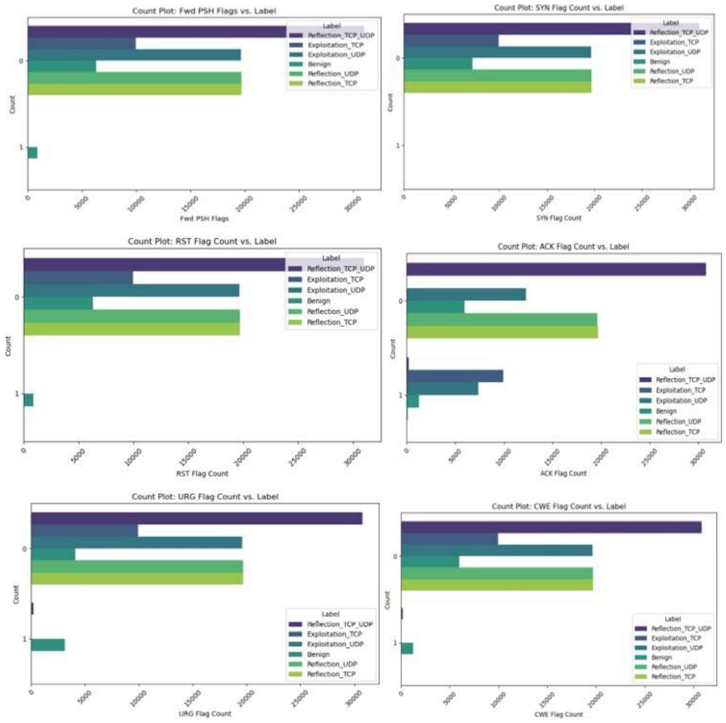


Figure 3: Count Plot of Various Flags with respect to Labels

An analysis is made between the count of various flags and their label as shown in Figure 3. The horizontal bar chart for the count plot for FWD PSH Flags vs. Labels provides a distribution of the forward PSH flags across different network traffic labels. The Reflection TCP has the highest count of forward PSH flags indicating a significant amount of traffic with these flags. The bar chart for the syn flag count vs labels give an insight into the distribution of the SYN flags for different traffic labels. The Benign traffic category has the highest count of SYN flags that represents a significant amount of the normal network traffic. For the bar chart, RST Flag Count vs labels give a distribution of RST (Reset) flags across different network traffic labels. Reflection TCP and Reflection UDP categories have the highest count of RST flags that tell a significant amount of reset activity in reflection-based network traffic. The count plot for ACK flag count vs Label provides the distribution of flags for different traffic labels. The count plot for URG flag count vs Label gives the distribution of URG (Urgent Flags) for different network traffic labels.

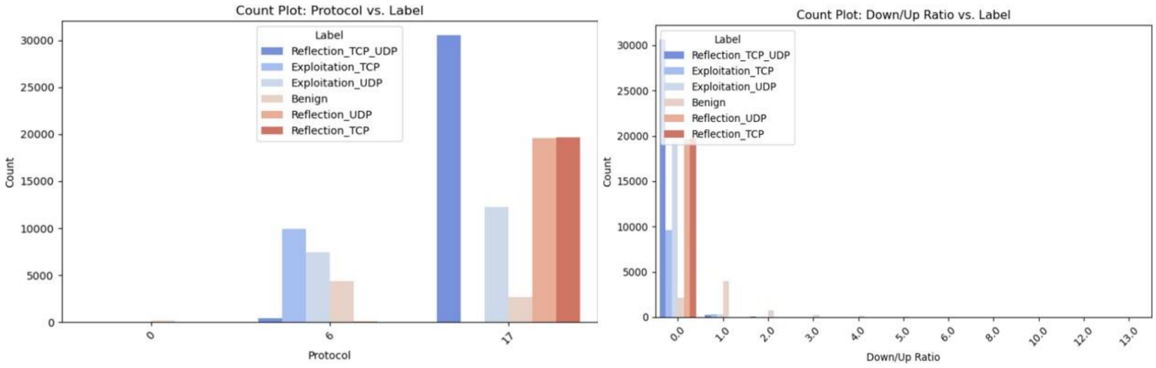


Figure 4: Count plot for Protocol and Down/Up Ratio with respect to Labels

Two bar plots are analyzed in Figure 4 which provides a count plot for columns with unique values protocol and down/up ratio. The bar chart for the protocol vs labels provides a distribution of different network traffic labels for three protocols (0, 6, and 17). Protocol 0 has the highest Benign count, indicating that most of the traffic in this protocol is normal and non-malicious. The bar chart for Down/Up ratios vs label provides insights into the distribution of the different categories based on Down/Up ratios. Reflection TCP category having the highest count which indicates a significant amount of traffic with a high Down/Up ratio.

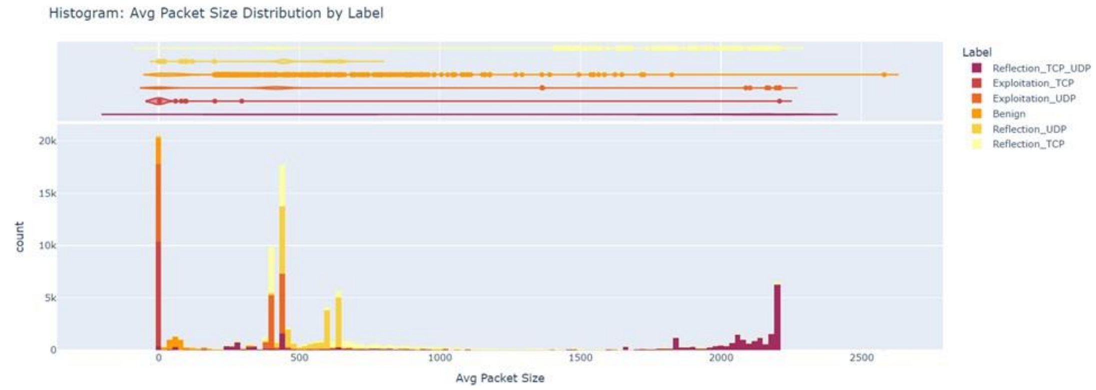


Figure 5: Average Packet Size Distribution by Label

Two bar plots are analyzed in Figure 5 which provides a count plot for columns with unique values protocol and down/up ratio. The bar chart for the protocol vs labels provides a distribution of different network traffic labels for three protocols (0, 6, and 17). Protocol 0 has the highest Benign count, indicating that most of the traffic in this protocol is normal and non-malicious. The bar chart for Down/Up ratios vs label provides insights into the distribution of the different categories based on Down/Up ratios. Reflection TCP category having the highest count which indicates a significant amount of traffic with a high Down/Up ratio.

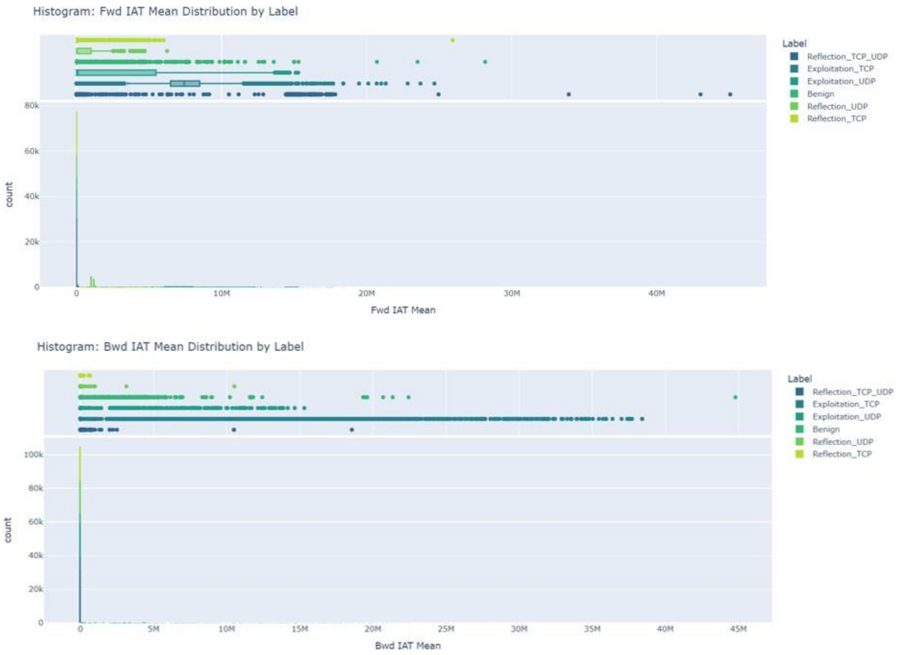


Figure 6: Distribution of FWD and BWD IAT Mean by Label

An analysis is carried out by plotting histogram plots for FWD IAT mean distribution by label and BWD IAT mean distribution by label shown in Figure 6. There are five types of labels categorized by histogram reflection UPD, reflection TCP, Benign exploitation TCP, and exploitation UDP.

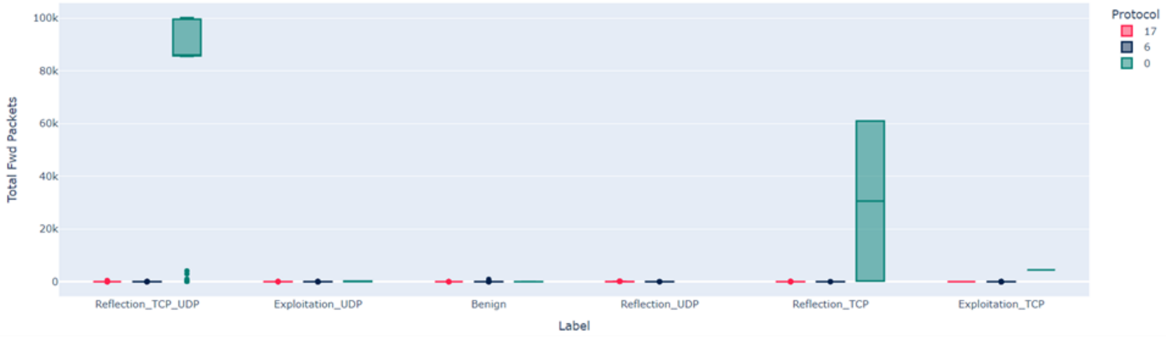


Figure 7: Total FWD Packets by Label and Protocol

A group box plot is plotted for the total FWD packets by label and protocol in Figure 7. The data is categorized into five labels reflection UDP, Exploitation UDP, Benign, Reflection ICMP, and Exploitation TCP. Each label was then divided into three protocols protocol 17 (red), protocol 6 (green), and protocol 0 (blue). A total find packet is represented by a y-axis ranging from 0 to 100k.

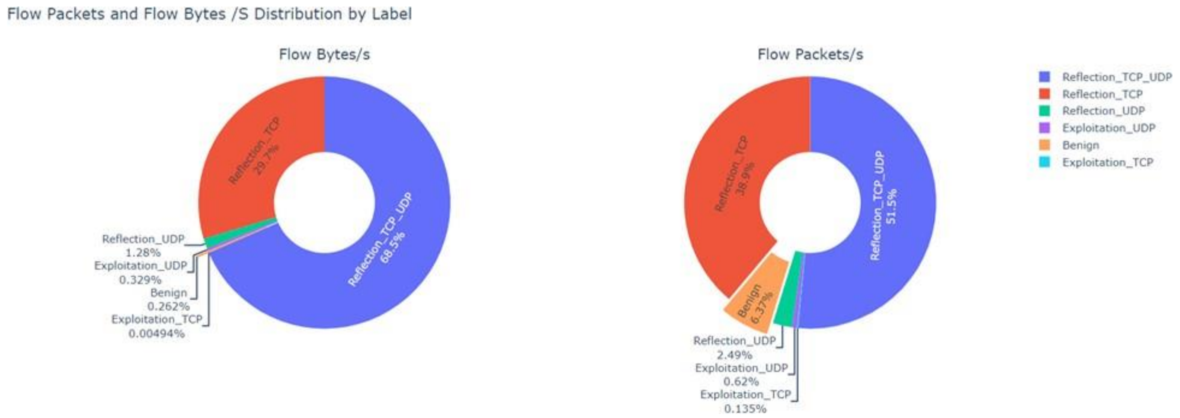


Figure 8: Flow Packets and Flow Byte/s distribution by label

A distribution chart is plotted as shown in Figure 8. It can be found that 95.99% of the benign type dominates the chart, 2.46% for accounts, and for flow packets/s distribution by label, the benign has the largest share with 93.91%.

3.4 Feature Engineering

Feature engineering is a technique in which new features are created, selected, or modified from the raw data to increase the performance of a machine-learning model. Features are input values that are given to the model for making predictions. By converting the raw data into meaningful features, the model learns patterns for making accurate predictions. The accuracy of the model can be significantly improved and the effectiveness of machine learning models by giving relevant information. Performing feature engineering simplifies the model and makes it easy to interpret and understand. In research, feature engineering is carried out by separating the target variable from the dataset in the machine learning workflow. In Figure 9, it can be seen that the target values are imbalanced with the highest number of classes being Reflection TCP UDP which can give an inaccurate result.

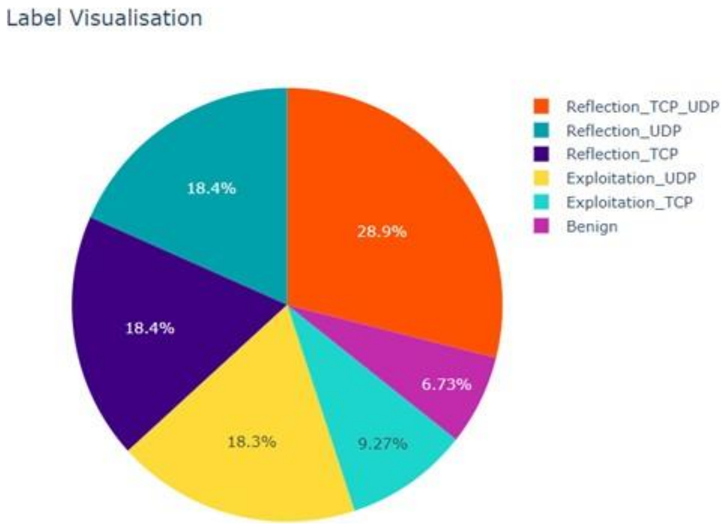


Figure 9: Label Visualization representing imbalanced data

To address this problem, a balanced dataset is created for the classes having a similar number of samples with the help of SMOTE on the data to generate the synthetic samples for the minority classes. Doing this prevents the machine learning algorithm from being biased for the majority of classes and the performance of the model is also increased, especially for cases when the minority class is of particular interest. After applying SMOTE, the classes are balanced as shown in Figure 9.

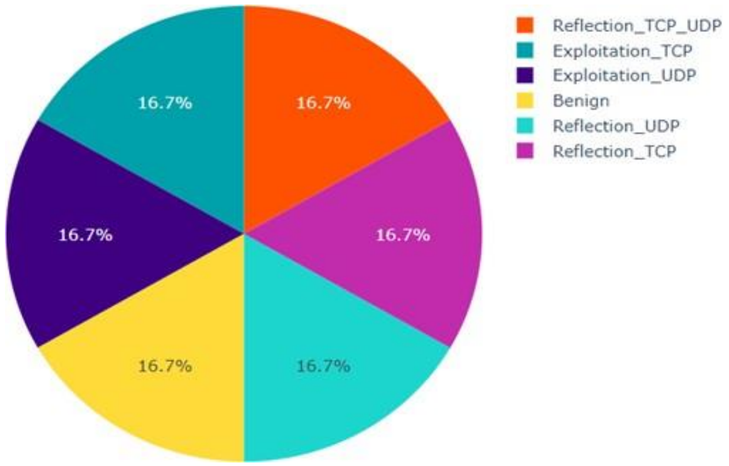


Figure 10: Label Visualization representing balanced dataset After SMOTE

After applying SMOTE, data is balanced as shown in Figure 10, the feature is scaled in the range [0,1] which is the crucial process in the machine learning model, especially when there is a different range in the features to ensure that no single feature dominates the learning process due to its scale. Following feature selection, the categorical features are one hot encoded to converted into 0 and 1, by one hot encoding the labels avoid ordinal interpretation for labels and ensure that the model does not assume any order across the categories. After performing one hot encoding, since the number of columns in the dataset is very high, it results in a high dimension of data. To reduce the dimension without losing the information can be achieved by implementing unsupervised algorithm PCA. The relevant features are selected with the help of PCA and Figure 11 shows the cumulative variance explained by the principal components which helps in identifying the number of components that are necessary to extract the desired amount of variance in the data.

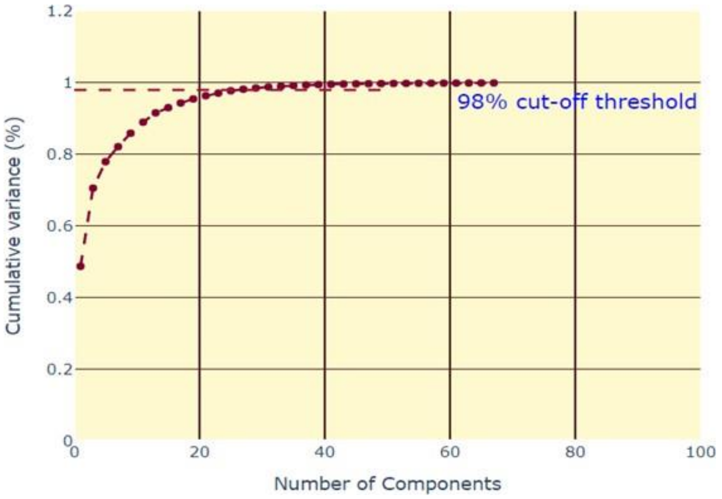


Figure 11: Principal Component Analysis (PCA) to get the optimal number of components

3.5 Model Training

When it comes to the part of making the prediction, model training plays an extremely crucial role. After the feature extraction has been done here, 15 features are taken into consideration using principal component analysis (PCA). After this, the data is interleaved into training and testing data with 30% of data allocated for testing. After splitting the data into training and testing sets, the training data and test data are reshaped into 2D for implementing the deep learning algorithms. Three deep learning algorithms are trained on the training set, the algorithms used in this research are CNN, RNN, and Autoencoders. CNN (Convolutional Neural Network) model with 8 layers and the total trainable parameter are 214 and the model is compiled with Adam optimizer and the categorical cross-entropy loss. The recurrent Neural Network (RNN) model with 8 layers 241 total parameters and 211 trainable parameters and the LSTM model is compiled with Adam optimizer and categorical cross-entropy loss and the Autoencoder Model has 22 layers with which 3 layers for encoder with a dropout, 4 layers for bottleneck and 3 layers for decoder with dropout layers and 5 layers for classification. The autoencoder model is compiled with Adam optimizer and categorical cross-entropy loss.

3.6 Model Evaluation

The final step after training the data on the algorithm is to test the performance of the algorithm in question. Hence, there is a strong need to assess the performance of the algorithm to know how better the algorithm is in dealing with the data. Four performance measurements are determined for the algorithms. The performance metrics used are accuracy (used when the classes are balanced and gives the proportion of instances that are correctly classified), precision – tells the proportion for the correct positive predictions, recall – tells the proportion of actual positives that are correctly identified and used when the rate of false negative is high and F1-score which is the harmonic mean of precision and recall. These metrics are measured by applying test data to the algorithm and its results are compared with the actual results of the test data. Therefore, it can be said that with the help of these metrics, the given algorithm is showing better performance.

4 Design Specification

To safeguard server infrastructure from Distributed Denial-of-Service (DDoS) attacks, we have developed and implemented a comprehensive DDoS Attack Detection Framework. The framework initially separates internet traffic into two categories: traffic originating from attack machines and traffic originating from normal users, which we call a client system. On the server end, the traffic is then examined by a deep learning-based anomaly detection method, which serves as the cornerstone of our framework. Using sophisticated algorithms, our system can identify the anomalies in traffic patterns and can effectively distinguish between normal behaviors and behaviors indicating potential DDoS attacks. The system classifies the anomaly according to a specific DDoS Attack Type after it has determined an anomaly, which is critical in deciding the best mitigation approach for the situation. The ability to decide whether to allow or block the incoming traffic based on the detected attack type lies with the mitigation strategy component of our framework. The process of this mitigation is carried out using various strategies such as rate limiting, IP-based blacklisting, or a variety of even more advanced techniques to minimize the impact of the attack. An alert system is also included that notifies administrators when an attack is detected or automatically applies specific responses. Administrators must decide what actions to take after an alert is sent, which may include applying rate limiting, blocking a specific IP, or other steps to mitigate the effects of an attack. The server which is the target of the DDoS attack changes settings as per the mitigation decisions to allow normal users of the service and block malicious traffic. The flow diagram of the DDoS Attack detection framework is shown in Figure 12

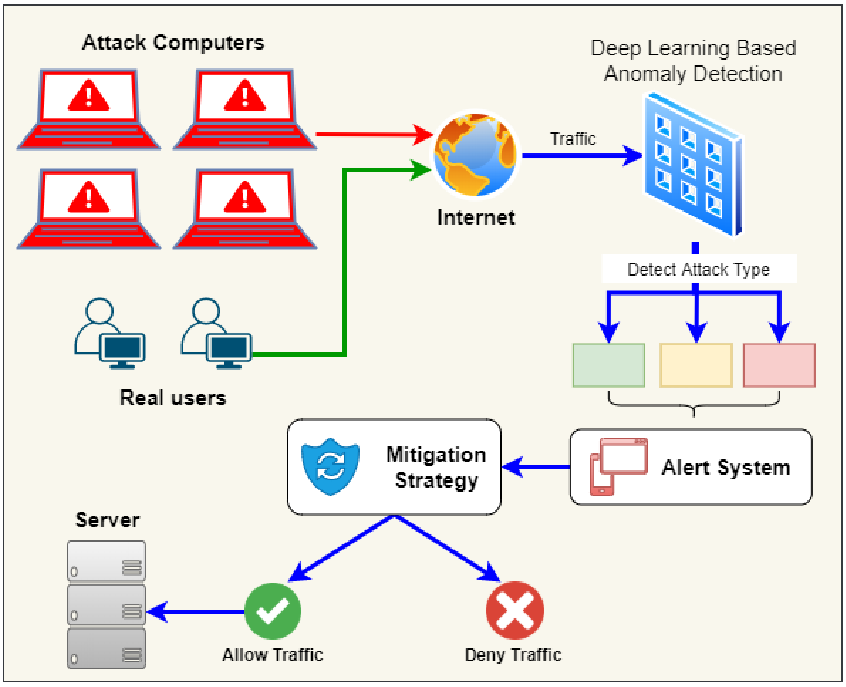
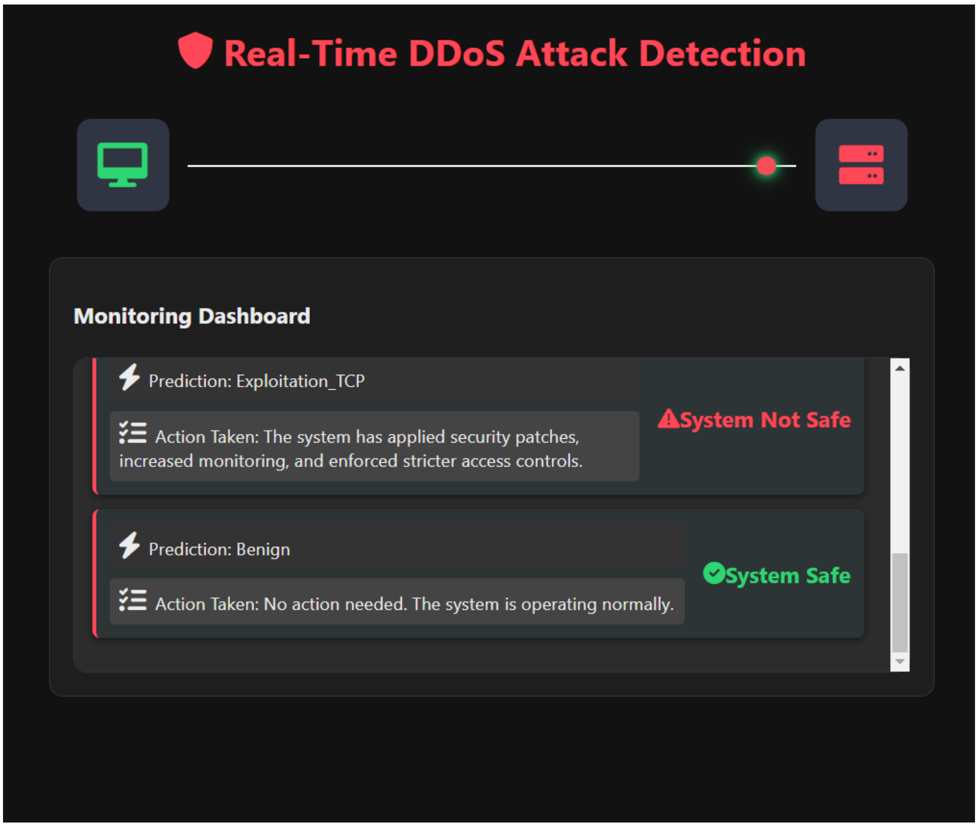


Figure 12: DDoS Attack Detection Framework

5 Implementation

We have implemented a DDoS Attack Detection Framework using Python in our research. We have employed various libraries representing different purposes. NumPy has been used for numerical operations and array manipulations, and the Pandas to manipulate and analyze data including the processes of handling as well as processing of datasets. Further, we employed Matplotlib to produce static visualizations allowing us to improve our understanding of data and model performance along with Seaborn which supports to production of improved statistical graphics and plots that are visually more appealing. We have also utilized Plotly to produce interactive plots along with dashboards allowing visualizing real-time data and results, Scikit-learn (sklearn) to implement machine learning algorithms and to evaluate model performance, and Imbalanced-learn (imblearn) to apply SMOTE to handle a class imbalance in the dataset through synthetic data generation. We have applied TensorFlow for building and training deep learning models, along with Keras functioning as a high-level API to assist in simplifying the model construction and training processes.

In addition, a web application was developed using the Python Flask framework following a client-server architecture. In the architecture, network packets are sent from the client to the server at regular intervals, where the server processes the network packets in real time with the deep learning model embedded to detect anomalies within the network system. The most optimal deep learning has been obtained after evaluating all model performances. Our system categorizes and detects all types of DDoS attacks and normal network traffic. When the DDoS attack is detected, the system suggests the appropriate mitigation plan based on the type of attack detected, or if the traffic is deemed normal, no action is taken and the network continues to function as normal. For the user interface design of the web application, we used HTML, JavaScript, and CSS. This web application efficiently detects the attacks in real time is deployed on the local system and demonstrates the practical implementation and readiness for deployment in larger environments. A screenshot of the web application, detecting the DDoS attack in real-time is shown in Figure 13.



Real-Time DDoS Attack Detection

Monitoring Dashboard

Prediction: Exploitation_TCP

Action Taken: The system has applied security patches, increased monitoring, and enforced stricter access controls.

System Not Safe

Prediction: Benign

Action Taken: No action needed. The system is operating normally.

System Safe

Figure 13: Web Application Detecting the DDoS Attack from Network System in Real-time

6 Evaluation

To perform the research, three algorithms that are distinct from one another are used. CNN, RNN and Autoencoder model of deep learning are the commonly used approaches. The research problem is divided into the categories of attacks Benign attacks, Reflection TCP UPD attacks, Exploitation TCP attacks, Exploitation UDP attacks, Reflection UDP attacks, and Reflection TCP attacks. The following assessment parameters are used including accuracy, precision, recall, and F1-score, all of these metrics are used to analyze the evaluation of each algorithm. Further, the performance of these algorithms is compared with each other to identify the most optimal model.

6.1 Evaluation Based on Accuracy

Accuracy defines the ratio of instances that are classified to the total instances that are available. It is used when the classes are balanced. The density of the links between the classes is the same or nearly equal on each side. The higher the values, it can be said that the model demonstrates better performance. In the experiment, the measure of each model concerning accuracy gives a quantitative analysis as to how well the model performs. The accuracy achieved through the proposed CNN model results in about 75.11%, where by the accuracy that has been realized by RNN is higher as compared to CNN stands at 96.40% However, the Autoencoder model provides a better inherent accuracy of

99.66%. Thus, the Autoencoder model leverages the highest accuracy compared to other algorithms. Figure 14 shows the comparative study of the model in terms of accuracy.

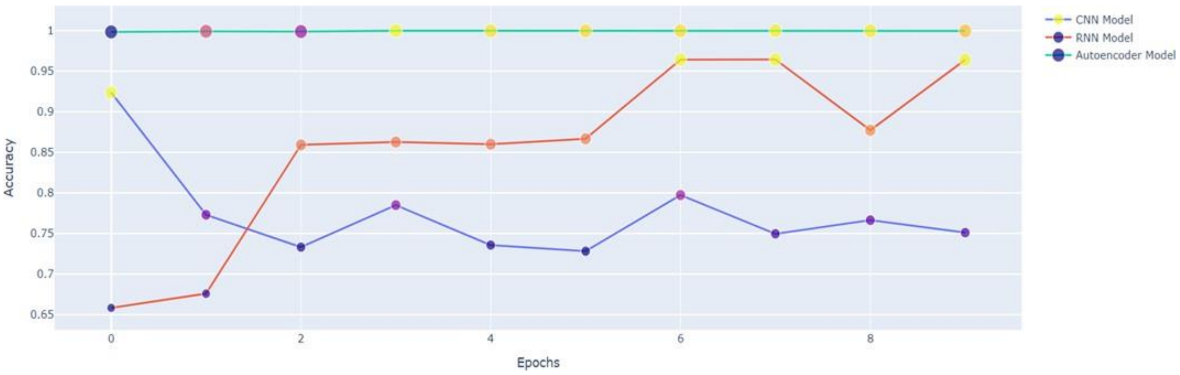


Figure 14: Comparative Analysis of Model Performance based on Accuracy

6.2 Evaluation Based on Precision

Precision defines the extent of positivity that is distinctive and is employed when the likelihood of a false positive outcome is high. As for the experiment, the CNN model yields a precision of 73.87% which is much less than the precision achieved by the RNN model which is 96.62%. The Autoencoder model in this research achieved a high level of precision, which stood at 99.97%. Of the three algorithms, the best average precision achieved was 99.97%. Based on the precision for positive predictions, the Autoencoder algorithm provides higher accuracy as we see from the performance comparison, Figure 15 represents a bar chart for the comparative analysis based on precision for the models.

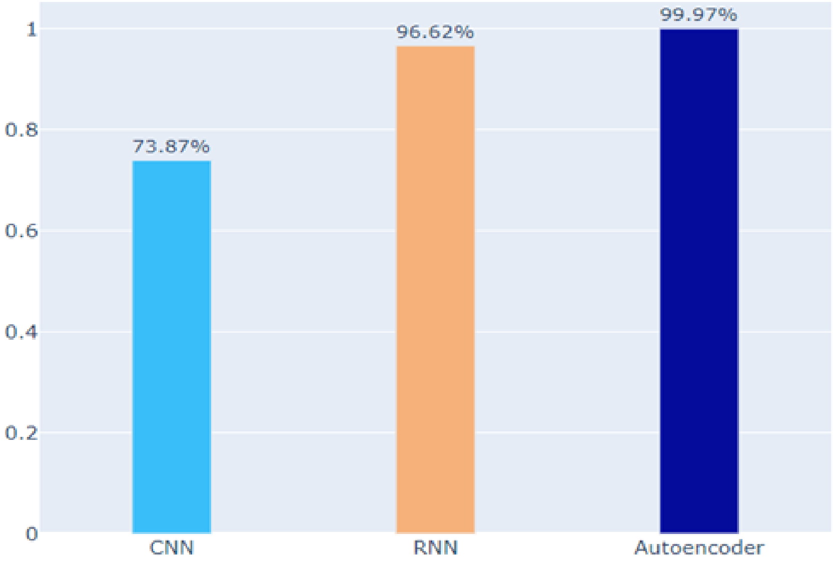


Figure 15: Comparative Analysis of Model Performance based on Precision

6.3 Evaluation Based on Recall

True Positive Rate or Recall or Sensitivity gives the percentage of actual positives that are correctly identified and recall is used when the false positive rate is high. In the experiment, to assess the models' performance of getting the positive instances, recall is used to get the insights. CNN model shows a recall of 75.11%. According to the RNN model, the recall is 96.41% which is higher than the CNN model. The Autoencoder model reaches a recall of 99.97% which means that it was better than two algorithms in identifying the positive instances. From the high recall performance of the Autoencoder model, it can be concluded that it is more beneficial in identifying the true positive cases and hence makes it a more reliable model in this approach to detect the very essential positives. In this aspect, figure 16 is displayed as a donut chart and it presents the recall comparison of the three algorithms.

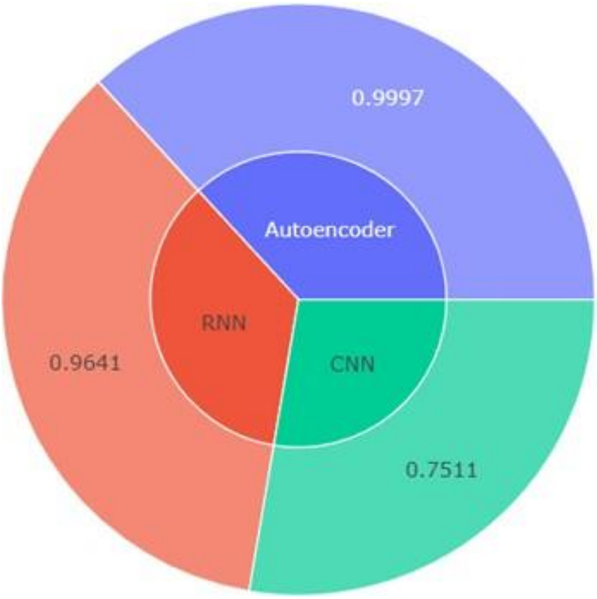


Figure 16: Comparative Analysis of Model Performance based on Recall

6.4 Evaluation Based on F1-Score

In this experiment, the F1-score, which is the harmonic mean of the precision or recall gives a single value for each that balances between them. The F1-score of the CNN model is 71.62%, which indicates that there is both the recall and precision of a balanced performance. The F1-Score given by the RNN model is 96.37% which is higher than the CNN model. Autoencoder gives superior performance in terms of F1-Score with a score of 99.97%. Figure 17, show a polar chart for the comparative analysis of the models.

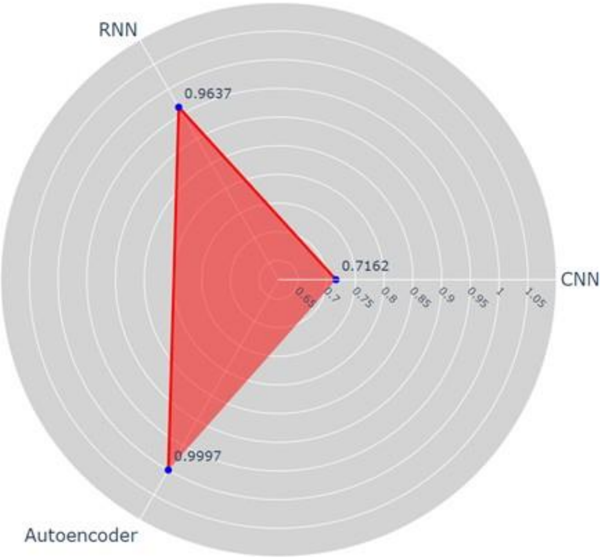


Figure 17: Comparative Analysis of Model Performance based on F1-Score

6.5 Discussion

In the research, three distinct algorithms were employed for the DDoS detection system, targeting five classes: benign attack, Reflection TCP UDP, Exploitation TCP attacks, Exploitation UDP attacks, Reflection UDP attacks, and Reflection TCP attacks. Reflection TCP UDP attack occurs when the attacker sends a TCP packet to the victim and obtains a TCP packet from the victim; Exploitation TCP occurs when the attacker sends a TCP packet to the victim and exploits the reply. Exploitation UDP occurs when the attacker sends a UDP packet to the victim and gets a reply to the UDP packet from the victim. Reflection UDP occurs when the attacker sends a UDP packet to the victim. The types of algorithms used in our research are Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Autoencoder where each was assigned the responsibility of identifying the different types of attacks. These models were evaluated using four key performance metrics including precision, recall, and F1-score. Evaluating the results obtained on all the metrics, the Autoencoder model yielded the best results with an accuracy of 99.97%, a recall of 99.97%, and the F1-Score as high as 99.97%. This shows that the Autoencoder was even more accurate in either identifying positive or negative samples, and this makes it the most accurate method of all the three. RNN model indicates a high readiness as competent in dealing with sequential data. The CNN model, although quite good at the job it was designed for, produced slightly lower accuracy on the data sets. These dissimilarities imply that one type of algorithm is stronger in one aspect than the other whereas the Autoencoder gives a higher overall performance in perceiving DDoS across multiple categories and therefore, would be recommended for real applications in cybersecurity. The autoencoder model will be saved and further will be utilized for predicting the different types of DDoS attacks in real-time based on which the web application also has been developed using the Flask framework.

7 Conclusion and Future Work

In the present day, with the increasing significance of data security, the swift and precise identification of cyber threats is of utmost importance. The number of cyber threats is on the rise, and attackers are employing more sophisticated methods to avoid detection by traditional security measures. To meet the challenges this presents, we have chosen to look at deep learning algorithms for threat prediction and threat prevention, with particular attention paid to improving data security. In this study, we implemented three types of deep learning model: CNN, RNN, and Autoencoders. Then, we evaluated the effectiveness of these models in identifying and preventing threats and improving the security of data. We created and evaluated the performance of the three deep learning models–CNN, RNN, and Autoencoders–for five threat classifications: Benign, Exploitation TCP, Exploitation UDP, Reflection UDP, Reflection TCP, and Reflection TCP UDP. By evaluating these models on standard performance metrics, including accuracy, precision, recall, and F1-score, the best-performing model was found to be the Autoencoder model.

The key contribution of the study is the creation of a web-based real-time application, that uses the Autoencoder model in a client-server architecture. Which not only identifies DDoS attacks but also delivers concrete strategies for mitigation and offers a realistic and deployable solution to improve cyber security. The real-time characteristic and the incorporation of deep learning models that are used in a user-friendly, working application is what makes this unique. Continuing, future work will look towards more advanced algorithms and implementing real-time monitoring systems, so that DDoS attacks can be recognised and acted upon quickly. Additional work could also consider lightweight security protocols, or exploring options to move security measures to the edges, which would protect from and rectify attacks at the source. Other critical attacks such as APT attacks are more dangerous than DDoS attacks, using deep learning algorithm framework can be designed to detect such attacks.

References

Abood, M. J. K. and Abdul-Majeed, G. H. (2024). Enhancing multi-class ddos attack classification using machine learning techniques, *Journal of Advanced Research in Applied Sciences and Engineering Technology* **43**(2): 75–92.
URL: *https://doi.org/10.37934/ARASET.43.2.7592*

Aktar, S. and Yasin Nur, A. (2023). Towards ddos attack detection using deep learning approach, *Computers & Security* **129**: 103251.
URL: *https://doi.org/10.1016/J.COSE.2023.103251*

Amitha, M. and Srivenkatesh, M. (2023). Ddos attack detection in cloud computing using deep learning algorithms, *International Journal of Intelligent Systems and Applications in Engineering* **11**(4): 82–90.
URL: *https://ijisae.org/index.php/IJISAE/article/view/3456*

Bahashwan, A. A., Anbar, M., Manickam, S., Al-Amiedy, T. A., Aladaileh, M. A. and Hasbullah, I. H. (2023). A systematic literature review on machine learning and deep learning approaches for detecting ddos attacks in software-defined networking, *Sensors*

23(9).
URL: *https://doi.org/10.3390/S23094441*

Behal, S., Kumar, K. and Sachdeva, M. (2017). Characterizing ddos attacks and flash events: Review, research gaps and future directions, *Computer Science Review* **25**: 101–114.
URL: *https://doi.org/10.1016/J.COSREV.2017.07.003*

Bravo, S. and Mauricio, D. (2019). Systematic review of aspects of ddos attacks detection, *Indonesian Journal of Electrical Engineering and Computer Science* **14**(1): 155–168.
URL: *https://doi.org/10.11591/IJEECS.V14.I1.PP155-168*

Fadlil, A., Riadi, I. and Aji, S. (2017). Review of detection ddos attack detection using naive bayes classifier for network forensics, *Bulletin of Electrical Engineering and Informatics* **6**(2): 140–148.
URL: *https://doi.org/10.11591/EEI.V6I2.605*

Hadi, T. H. (2024). Deep learning-based ddos detection in network traffic data, *International Journal of Electrical and Computer Engineering Systems* **15**(5): 407–414.
URL: *https://doi.org/10.32985/IJECES.15.5.3*

Kapko, M. (2024). Ddos attack traffic surged in 2023, cloudflare finds — cybersecurity dive.
URL: *https://www.cybersecuritydive.com/news/ddos-attacks-surge-cloudflare/704011/*

Kumar, D., Pateriya, R. K., Gupta, R. K., Dehalwar, V. and Sharma, A. (2023). Ddos detection using deep learning, *Procedia Computer Science*, Vol. 218, pp. 2420–2429.
URL: *https://doi.org/10.1016/J.PROCS.2023.01.217*

Kumari, K. and Mrunalini, M. (2022). Detecting denial of service attacks using machine learning algorithms, *Journal of Big Data* **9**(1): 1–17.
URL: *https://doi.org/10.1186/s40537-022-00616-0*

Li, Q., Huang, H., Li, R., Lv, J., Yuan, Z., Ma, L., Han, Y. and Jiang, Y. (2023). A comprehensive survey on ddos defense systems: New trends and challenges, *Computer Networks* **233**.
URL: *https://doi.org/10.1016/J.COMNET.2023.109895*

Mustapha, A., Khatoun, R., Zeadally, S., Chbib, F., Fadlallah, A., Fahs, W. and El Attar, A. (2023). Detecting ddos attacks using adversarial neural network, *Computers & Security* **127**.
URL: *https://doi.org/10.1016/J.COSE.2023.103117*

Najar, A. A. and Manohar Naik, S. (2024). Cyber-secure sdn: A cnn-based approach for efficient detection and mitigation of ddos attacks, *Computers & Security* **139**.
URL: *https://doi.org/10.1016/J.COSE.2024.103716*

Saini, P. S., Behal, S. and Bhatia, S. (2020). Detection of ddos attacks using machine learning algorithms, *Proceedings of the 7th International Conference on Computing for Sustainable Global Development (INDIACom 2020)*, pp. 16–21.
URL: *https://doi.org/10.23919/INDIACOM49435.2020.9083716*

Salih, A. A. and Abdulrazaq, M. B. (2024). Cybernet model: A new deep learning model for cyber ddos attacks detection and recognition, *Computers, Materials & Continua* **78**(1): 1275–1295.
URL: *https://doi.org/10.32604/CMC.2023.046101*

Tekleselassie, H. (2021). A deep learning approach for ddos attack detection using supervised learning, *MATEC Web of Conferences* **348**: 01012.
URL: *https://doi.org/10.1051/matecconf/202134801012*

UNB (2019). Ddos 2019 — datasets — research — canadian institute for cybersecurity.
URL: *https://www.unb.ca/cic/datasets/ddos-2019.html*

Wakamiya, S., Li, C.-T., Huang, C.-T., Becerra-Suarez, F. L., Fernández-Roman, I. and Forero, M. G. (2024). Improvement of distributed denial of service attack detection through machine learning and data processing, *Mathematics* **12**(9): 1294.
URL: *https://doi.org/10.3390/MATH12091294*

Wei, Y., Jang-Jaccard, J., Sabrina, F., Singh, A., Xu, W. and Camtepe, S. (2021). Ae-mlp: A hybrid deep learning approach for ddos detection and classification, *IEEE Access* **9**: 146810–146821.
URL: *https://doi.org/10.1109/ACCESS.2021.3123791*

Yoachimik, O. and Pacheco, J. (2024). Ddos threat report for 2023 q4.
URL: *https://blog.cloudflare.com/ddos-threat-report-2023-q4*