International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 9 Issue 6 June - 2025, Pages: 93-105

Design And Implementation Of A Machine Learning Based DDoS ATTACKS Detection And Mitigation System For Network Security

Ibeh Sylvarine Chinasa[1], Ike Joseph Mgbemfulike[1] and Ogochukwu C. Okeke[1].

[1]Department of Computer Science, Faculty of Physical Sciences, Chukwuemeka Odumegwu Ojukwu University, Uli, Anambra State, Nigeria.

Abstract: *With the rapid growth of internet-connected systems, Distributed Denial-of-Service (DDoS) attacks have become one of the most prevalent and disruptive threats to network security. Traditional rule-based intrusion detection systems often struggle to identify evolving and sophisticated DDoS attack patterns in real-time. This project presents the design and implementation of a machine learning-based system for the detection and mitigation of DDoS attacks. By leveraging traffic flow features such as packet rates, IP entropy, and protocol usage, the system utilizes supervised learning algorithms—specifically Random Forest and Support Vector Machine (SVM)—to classify traffic as benign or malicious. Real-time packet capturing and feature extraction are integrated into a detection pipeline, which enables timely and automated responses to identified threats. Upon detection, mitigation is carried out through dynamic firewall rules and traffic rate limiting. Experimental results using benchmark datasets such as CICDDoS2019 demonstrate high detection accuracy and low false positive rates, validating the effectiveness of the proposed system. This approach not only enhances the responsiveness of network defense mechanisms but also provides a scalable solution to adapt to emerging attack vectors.*

Keywords: Machine Learning (ML), DDoS (Distributed Denial of Service, Cyber security, Network Security, Intrusion Detection System (IDS), Anomaly Detection, Mitigation Techniques, Traffic Classification, Cybersecurity.

1. Introduction

In today's digitally interconnected world, the security and availability of networked systems are critical to both public and private sector operations. Among the various forms of cyber threats, **Distributed Denial-of-Service (DDoS) attacks** stand out due to their ability to flood networks with illegitimate traffic, rendering online services inaccessible to legitimate users. These attacks have grown not only in scale but also in sophistication, often bypassing traditional intrusion detection systems (IDS) and overwhelming network defenses.

As the limitations of static, rule-based security mechanisms become increasingly evident, there is a growing need for more **adaptive, intelligent, and automated solutions**. In this context, **machine learning (ML)** has emerged as a powerful tool for cyber security, offering the ability to detect previously unseen attack patterns by learning from historical and real-time data. ML algorithms can analyze vast volumes of network traffic to differentiate between normal behavior and potential threats, making them particularly suitable for the detection of DDoS attacks.

This study focuses on the **design and implementation of a machine learning-based system** for the **detection and mitigation of DDoS attacks**. The proposed system captures live network traffic, extracts relevant features, and applies trained ML models to classify traffic as either benign or malicious. Upon detecting a DDoS threat, the system responds automatically by initiating mitigation actions such as traffic filtering or rate limiting to protect the network infrastructure.

The significance of this study lies in its ability to provide **real-time protection** against DDoS attacks using a scalable, data-driven approach. The system is evaluated using publicly available datasets and real-time testing environments to demonstrate its effectiveness in detecting various DDoS attack vectors with high accuracy and low false positive rates.

Through this work, we aim to contribute a practical, intelligent, and deployable solution to the ever-growing challenge of securing networks against DDoS attacks, and to highlight the transformative role that machine learning can play in modern network defense systems.

2. Related works

In Mamoon et al. (2023) work, machine learning was used to discover the DDoS attack and know its type to be aware of it and take the necessary measures for that. They used the CICDDoS2019dataset. They algorithm they used are Random Forest, Decision Tree, SVM, Naive Bayes, and xgboost which were used to train and test the data from the datasets. In their result, Random Forest algorithms had the highest level of accuracy (99.95426%).

Alzahrani and A.Alzahrani (2023) proposed a deep learning approach for identifying and thwarting flood attacks, also known as DoS-based Hello on the IoT healthcare network. They used the Deep Belief Network (DBN) model to confirm this kind of

attack, which entailed sending plenty of Hello packets to slow down the network. The bypass-linked attacker update-based rider optimization algorithm (BAU-ROA) is a tool that the DBN approach has used to produce a variety of useful outcomes and function even better.

The development of the high-performing optimization technique known as BAU-ROA, a metaheuristic algorithm with a simple calculation methodology and fewer computing parameters, was done to enhance the execution of ROA. The BAU-ROA algorithm has been shown to perform better than other optimization algorithms when it comes to the DBN operational procedure in experiments (Natabine, et. al.,2022).

According to IoT application and design, the authors of (Kushwah et al., 2021) provided a thorough analysis of recent and earlier studiesin IoT traffic characterization. In their survey, the main focus on traffic characterization for security issues has been highlighted as the key attention of the articles in IoT.The accuracy, precision, recall, and F1 score of four MLalgorithms i.e., DT, KNN, NB, and gradient-boosting (GRB)classifiers were compared in their study, along with the performance of each approach overall. They made use ofthe BoT-IoT dataset. DT and GRB performed better in terms of accuracy, according to the performance evaluation findings of their study. The IoT networks' greater security will be aided by these impressive achievements.

Kamber et al., (2022) proposed a hybrid machine learning-based technique. Black hole optimization and the extreme learning machine(ELM) model are combined to implement the suggested method. To test the effectiveness of our suggested technique, several experiments have been carried out using four benchmark datasets: NSL KDD, ISCX IDS 2012, CICIDS2017, and CICDDoS 2019. The accuracy is 99.23%, 92.19%,99.50%, and 99.80% using NSL KDD, ISCX IDS 2012,CICIDS 2017, and CICDDoS 2019, respectively. It is also done to compare the performance of the proposed method with existing methods based on ELM, backpropagation ANN, artificial neural network (ANN)trained with blackhole optimization, and other cutting-edge methods (Sharafaldin, et. al., 2019).

Alireza et al., (2021) presented a DDoS detection model based on data mining and machine learning techniques. They used the CICDDoS2019 dataset, they experimented with the following machine learning algorithms: Naïve Bayes, SVM, KNN, Random Forest,XGBoost, and AdaBoost. It is discovered that AdaBoost and XGBoost were extraordinarily accurate and correctly predicted the type of network traffic with 100% accuracy.

(Lohachabet. al., 2018) added a well-known DDoS dataset called CICDoS2019 that wouldi mprove the accuracy of DDoS attack identification. The DDoS dataset has also undergone preprocessing utilizing two major methods to extract the most pertinent information. TheDDoS dataset will be used with four distinct machine-learning models. The Random Forest machine learning model, with an improvement over recently developed DDoS detection systems, provided the best detection accuracy with(99.9974%), according to the results of actual testing.

Hudaib et al., (2014) applied the three-level application layer architecture for detecting DDoS attacks. The first level is in charge of choosing the samples' best attributes and categorizing the traffic as either benign or malicious; the second level is made up of a hard-voting classifier to determine if the DDoS source is UDP, TCP, or mixed-based. Last but not least, the DDoS type that best suits the attack is aligned at this level. The accuracy score, precision, and time were employed as the model performance metrics in the approach's validation on the CIC-DDoS2019 dataset. The suggested architecture significantly outperforms the currently used machine learning (ML) methodologies in categorizing application-layer DDoS attacks both binary and multiclass(Alonso et al., 2008).

She et al. (2017) proposed a model to distinguish normal users from botnets which are used to perform DDoS attacks on the application layer based on seven extracted features from user sessions. They used a one-class SVM algorithm on their gathered Dataset and concluded their model was effective application layer DDoS detection.

Wankhede and Kshirsagar et al. (2018) proposed a model to detect DoS attacks based on machine learning and neural networks, then tried to maximize their model's accuracy compared to similar detection models by setting the optimum value of parameters. They achieved an accuracy of 99.95%via Random Forest algorithm with 500 trees and 50% training data set on CIC IDS 2017 dataset.

Hoon et al., (2018) examined and analyzed different supervised and unsupervised machine learning algorithms which were being used for DDoS detection by previous researchers. They have also found the optimum value of hyper parameters that could maximize the accuracy of algorithms. The main contribution of their research is the introduction of a parameter called P(A), which is used as at threshold for better decision-making as to the training time. They have tested different algorithms on NSL-KDD Data se tand found out that the supervised algorithms like RandomForest, Gradient Boost, and Naïve Bayes have better performance in terms of accuracy and training time. CORRECTION STOPPED HERE!!!!!!!!!!!!!!!!!!!!

Ates et al., (2019) proposed a DDoS detectionsystem based on request packet header relations. Theyperformed experiments on real extracted data and the Caidadataset and used Entropy and Modularity concepts and theSVM algorithm. They found out that the higher accuracy isachieved by using the Entropy concept in UDP connectionsand Modularity concepts in TCP connections.

Dong and Sarem (2019) also proposed two newalgorithms, DDAML and DDADA, based on KNN and the degree of attack concept. They gathered their Dataset from a simulation environment and generated DDoS traffic withhping3, and tested their proposed algorithms as well as other traditional machine learning algorithms like SVM, KNN, andNaïve Bayes. After

comparing the results of ROC curves,they found out that their proposed algorithms have better performance than the existing ones.

Sumathi and Karthikeyan (2018) compared different traditional and hybrid machine-learning algorithms. Theyhave tested these algorithms on KDDcup99 and DARPFdatasets and found that Decision Trees and Fuzzy C-Meansperform better than the others. Fuzzy C-Mean algorithmcould detect DDoS traffic with an accuracy of 98.7% andwith a detection time of 0.15 seconds.

Ajeetha and Pryia (2019) developed a DDoS detection system based on machine learning techniques and traffic flow traces. They have tested Naïve Bayes andRandom Forest algorithms on gathered datasets from SansandIsna and discovered that the Naïve Bayes algorithm withan accuracy of 90.90% is more accurate than the RandomForest algorithm with 78.18% accuracy.

Wehbi et al. (2019) reviewed the related studies on DDoS detection in the IoT environment and then proposed three new approaches using SVM, KNN, LPA, and QDAalgorithms and tested these approaches on CAIDA, 1999DARPA Intrusion Detection Dataset, and their simulated environment. Their contribution was a novel classification for feature extraction and proposed a seven-layer sequential model for DDoS detection. They have also introduced two new criteria for preventing the wrong detection of normal traffic as DDoS traffic, which is a common phenomenon from machine learning-based DDoS detection. Finally, they discovered that all three proposed approaches recorded acceptable performance, and Random Forest was the most accurate algorithm with an accuracy of 99.99%.

Polat et al., (2019) proposed a DDoS detectionsystem based on data mining techniques and machinelearning algorithms. They tested different machine learningalgorithms on this system and KDDCUP99 and comparedthem in terms of speed and accuracy. They empirically foundthe optimum value of some hyperparameters like 10 forCross-Validation Ratio and 66% of Dataset for trainingmodel size. Based on this research, the J48 algorithm has thehighest success rate of correct DDoS attacks detection.

Ibrahim et al., (2021) proposed amultilayered framework using machine learning algorithmsto detect Botnets that are used to perform DDoSattacks. They used a new approach for feature extraction,classification, and hyperparameter setting and tested KNN,SVM, and MLP algorithms on CTU-13 Dataset with their proposed framework. They discovered that, unlike previousresearchers' suggestions, the Oversampling technique could not improve the accuracy of algorithms. KNN algorithm recorded the highest accuracy of 91.51% inside their proposed framework.

Dhamor et al., (2021) worked on DDoS detection onIoT devices. First, they used a new approach for datapreprocessing on the CICDDoS2019 Dataset. Then theyevaluated the performance of different machine learningalgorithms for detecting DDoS traffic on their preprocessedDataset. They ultimately discovered that machine learningtechniques are effective for detecting DDoS attacks, andRandom Forest, with an accuracy of 99.24%, was the mostaccurate algorithm among the tested algorithms.

Hezavehi and Rahmaniet al. (2020) proposed an anomaly-based detection of DDoS attack methods in cloud environment using a third party auditor (TPA). A TPA along with DDoS attack detection capabilities called third party auditor notification generator (TPANG). The proposed detection frameworks combined a third party auditor notification generator along with notification of detection is called TPANGND.

Kushwah, et al. (2020) proposed a new method for detecting DDoS attacks in cloud computing environment. The new detection method is developed based on voting ELM (VELM) [4]. Here NSL-KDD dataset and ISCX intrusion detection dataset are used. It has been shown that proposed system gives better accuracy than other systems built based on back propagation ANN, ANN trained with black hole optimization, ELM, random forest and, Adaboost.

Kushwah, et al. (2019) presents new DDoS attack detection model by using ELM. Here the NSL-KDD dataset used for experimentation. The proposed detection model produces high detection rate and takes less computation time.

Idhammad et al. (2018) presented a new detection method of HTTP DDoS attacks in a cloud environment. The proposed detection method performs based on two ensemble learning algorithms such as Information Theoretic Entropy (ITE) and RF. A time-based sliding window technique is used to calculate the entropy of the feature of network header of the incoming traffic signals. CIDDS-001 (Coburg Intrusion Detection Dataset) is an up-to-date labelled flow-based dataset) in a Cloud environment based on Open Stack platform. The classification tasks are produce when the expected entropy exceeds its usual range the pre-processing.

Rawashdeh et al. (2018) proposed an anomaly intrusion detection technique in the hypervisor layer to depress DDoS performance between virtual machines. The proposed detection method is developed by the evolutionary neural network. The evolutionary neural network is incorporates the particle swarm optimisation (PSO) with neural network for DDoS attack detection and classification of the traffic data [3]. Here most previous research used KDD CUP 99 and NSL-KDD datasets to evaluate their approaches. On the other hand, the dataset only handles the traffic that exchanges between VMs, so the traffic that comes from an outside host machine could be studied in future work.

Sahiet al. (2017) developed a new classification based detecting system and preventing DDoS TCP flood attacks in public clouds environment. A new developed DDoS detection International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-7, Issue-8, August 2021 ISSN: 2395-3470 www.ijseas.com 358 method presents a solution to protect the stored records by classifying the incoming packets and building a decision according to the classification outcome. Wireshark network analyzer used to capture the flood attack. The proposed detection methods identify and establish whether a packet is regular or created from an attacker during the prevention phase.

Waniet al. (2019) presents a new detection algorithm based on SVM. Out of the three algorithms used SVM shows the better results in terms of accuracy, recall .precision, specificity and f measure closely followed by Random Forest. The datasets are carried out on the own cloud environment using Tor Hammer attacking tool.

He et al. (2017) presents a new DDoS attack detection model on the source side in the cloud environment based on machine learning approaches. Extract statistical features of four DDoS attacks and launch real attacks in lab settings for evaluation. This detection scheme statistical information from both the virtual machine and the hypervisor to avoid network packages from being sent out to the exterior system.

3. Research Methodology:

Several studies in the existing literature have analyzed DDoS attacks and contributed various protection mechanisms. The most broadly utilized defense methods are identifying and mitigating DDoS attacks, traffic separation, and trace-back the DDoS source. DDoS detection solutions are effectively separating typical streams of activity from unusual streams of activity. Traffic separation solutions obstruct substantial movement, while trace-back mechanisms must be compelling under sponsored activity performed for the most part after the assault. A large portion of DDoS identification systems has constrained achievements considering the accompanying difficulties like attacking frequently used legit requests to overload the target itself, making it difficult to distinguish an attack movement from normal activity and secondly quick ongoing recognition is troublesome due to the enormous measure of information associated with a computer network. Therefore the critical challenging concerns in identifying DDoS attacks are firstly distinguishing a genuine and sufficient selection of features that can be used to construct efficient models for differentiating DDoS attacks from normal traffic and secondly assessing the viability of the various machine-learning approaches employed in the discovery process.

Most of the existing models adopted statistical approaches which can be used to detect suspicious patterns in resource utilization in response to DDoS attacks. The issue with statistics-based identification is that it is not conceivable to discover the typical network packet distribution and it must be reproduced as a uniform distribution. However, obtaining fundamental characteristics from a massive network is critical for modeling network behaviors that are distinct from normal traffic.

3.1 Data set

Each and every one of the researchers that worked on the related area used data as pertain problem and the direction the researchers aimed to achieve, and as such some of the researchers used some trajectory dataset, mobile sensor data, data extracted from Google, some also used simulated dataset while some adopted the already ascertained dataset.

3.2 Data set

The dataset used in this work were extracted from kaggle containing (200000). This dataset consists of feature and instances. The feature i.e., class value has two possible values: normal traffic (benign), and DDOS attack which are nothing but class labels.

Table 3.1: The description and features of the different events

Features	**Description**
Normal traffic (benign)	This label indicates normal, non-malicious network traffic. Packets labeled as `Benign` represent routine communications with no threat to network security. This traffic is generally safe and expected in regular network activity.
DDoS attack	This label identifies network traffic associated with a Distributed Denial of Service (DDoS) attack that primarily utilizes the ACK (Acknowledgment) flag and PSH flag in TCP packets. The PSH flag is used to request immediate data transmission, while the ACK flag acknowledges receipt of previous packets. This type of attack aims to congest the target network and slow down or prevent legitimate traffic from being processed

3.3 Data Cleaning and Pre-processing

The preprocessing was done after data collection, the data collected were categorized into their respective classes, for each category, we have a good number of data prepared manually with labeling tools. The labeling tool was written with the python scripts. The effort is to make sure that, the preprocessed data meet the requirement for further analysis when applied to the machine learning algorithms. In the process of preparing the dataset, columns will be removed or retained based on their relevance to model performance and privacy concerns.

3.4 Filtering users

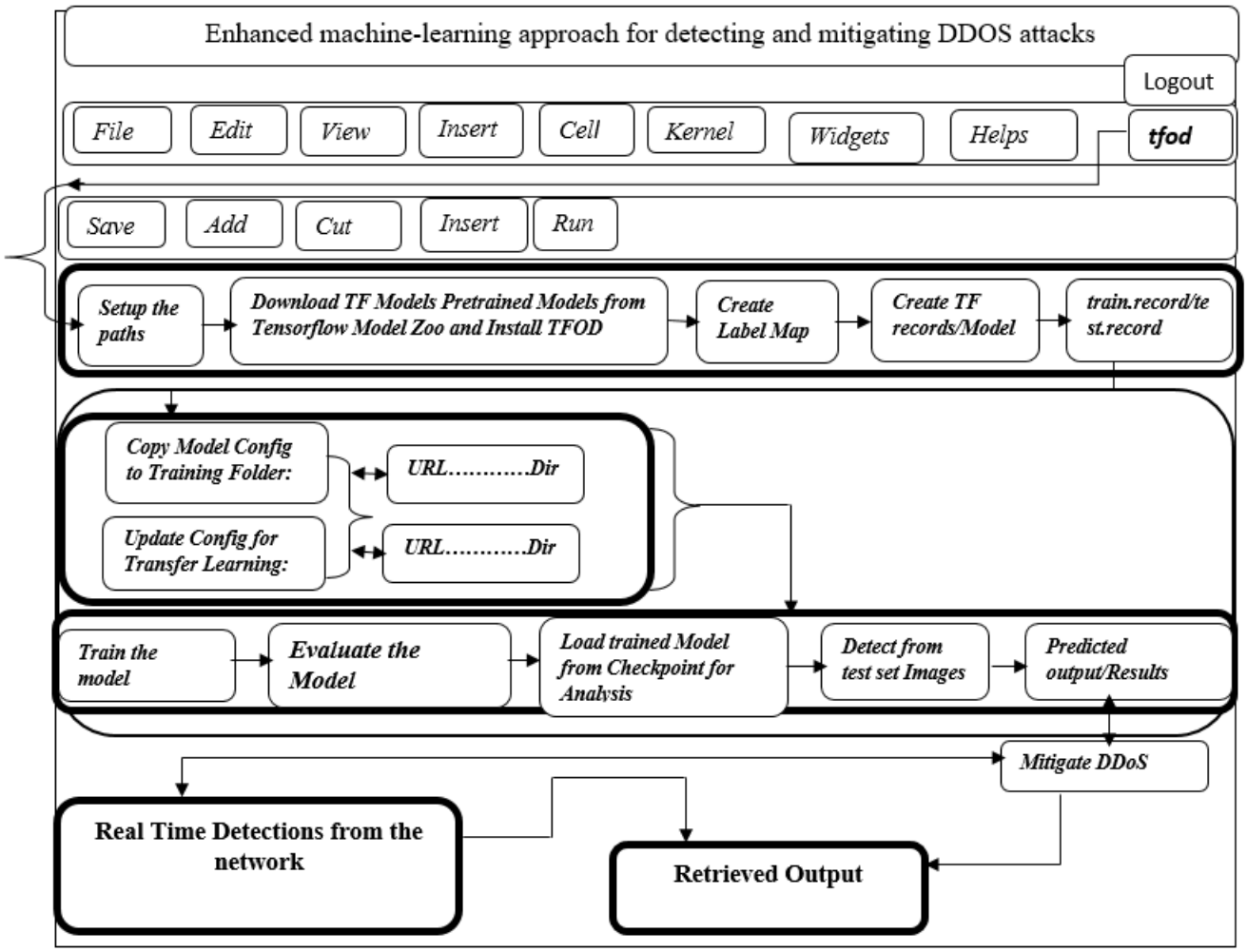
The amount of data that was extracted varied, this can have a negative effect on the learning of the machine learning algorithm. One solution to this was to remove data that do not comply with some constraints. A filter on each data event was created, removing data and their events when the number of sequence of events is below a certain threshold. This way data that goes against some constraints while collecting the data were removed.

The filtering techniques goes a long way of transforming the sample dataset into sizeable format by means of applying feature scaling and normalization to fit in the training model and as well to remove the missing value. This approach is done with the principle of applying statistical model such mean, mode, and standard deviation in order to transform the training data for analysis.

4. System Implementation:

System design is the process of designing the architecture, components, modules, interfaces and data for a computer system to satisfy s-pecified requirements. It is the process of defining and developing a system to satisfy specified requirements organization or individual as the case may be. The major objective of this work is to develop an enhanced machine-learning approach for detecting and mitigating DDOS attacks in network environments.

4.2 Control Centre/Main Menu



RESULTS AND DISCUSSION:

The system was tested with dataset collected via kaggle for DDoS attack which was basically used by the software. It was divided into two categories. If the test data was inadequately designed, the test inputs will not cover all-possible test scenarios, which will impact the quality of the application under test. The proposed system test data contained the following sample: sample dataset, model training sample data set, training and testing, training normal and DDoS attack and testing them which are in data folders. 20% of the sample dataset was used to evaluate the performance for testing the CNN and RF. Using test data, one can verify the expected result and the software behavior.

4.4 Actual Test Result versus Expected Test Result

Table 4.4 Actual test result and expected test result for RF

Attribute value	**Predicted class**
Actual Input outcome: 1	Predicted outcome: 1
Actual Input outcome: 1	Predicted outcome: 1

Actual Input outcome: 0	Predicted outcome: 0
Actual Input outcome: 0	Predicted outcome: 0
Actual Input outcome: 1	Predicted outcome: 1
Actual Input outcome: 1	Predicted outcome: 1
Actual Input outcome: 1	Predicted outcome: 1
Actual Input outcome: 1	Predicted outcome: 1
Actual Input outcome: 0	Predicted outcome: 0
Actual Input outcome: 0	Predicted outcome: 0
Actual Input outcome: 0	Predicted outcome: 0
Actual Input outcome: 1	Predicted outcome: 1
Actual Input outcome: 0	Predicted outcome: 0
Actual Input outcome: 0	Predicted outcome: 0
Actual Input outcome: 1	Predicted outcome: 1
Actual Input outcome: 1	Predicted outcome: 1

Table 4.5 Summary Actual Test Result and Expected Test Result

Actual Test Result	Expected Test Result
Normal	RF predicted accurate, CNN detected.and converged well
DDoS attack	RF predicted accurate, CNN detected and converged well

4.6. Performance Evaluation

The evaluation was done on Jupyter notebook and Scikitlearn. Jupyter notebook was used to evaluate the Convolutional Neural Network where 80% of the dataset was used for training and 20% for testing. Scikitlearn was used to evaluate Random forest where 80% of the dataset was used for the training and 20% was used for testing.

4.6.6.1 Model Evaluation for CNN

Table 4.7 CNN Classification Report of DDoS for model evaluation metrics

	A	B	C	D	E
1	Class	Precision	Recall	F1-Score	Support
2	0	1	1	1	100000
3	1	1	1	1	100000
4	Accuracy			1	200000
5	Macro Avg	1	1	1	200000
6	Weighted Avg	1	1	1	200000

Table 4.8 provided the classification details of CNN model with the acurray of 100% on the evaluation of image test set; CNN classifier is based on structural risk minimization. Figure 4.16 presented the result of confusion matrix which captured all the details in the testing data.

Contingency table/confusion matrix for CNN

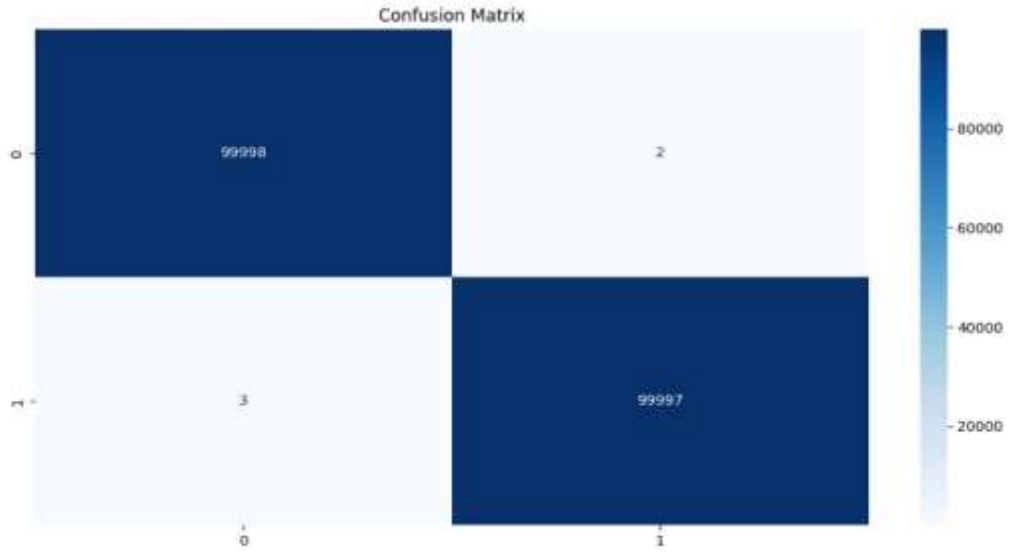


Figure 4.16 Contingency table/confusion matrix for CNN

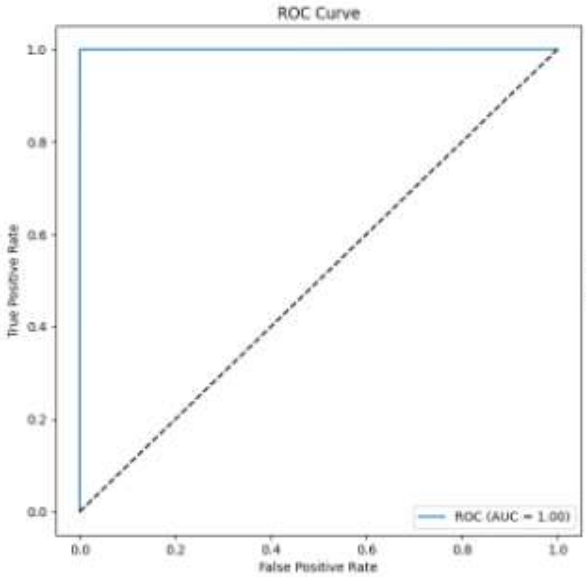


Figure 17 ROC curve

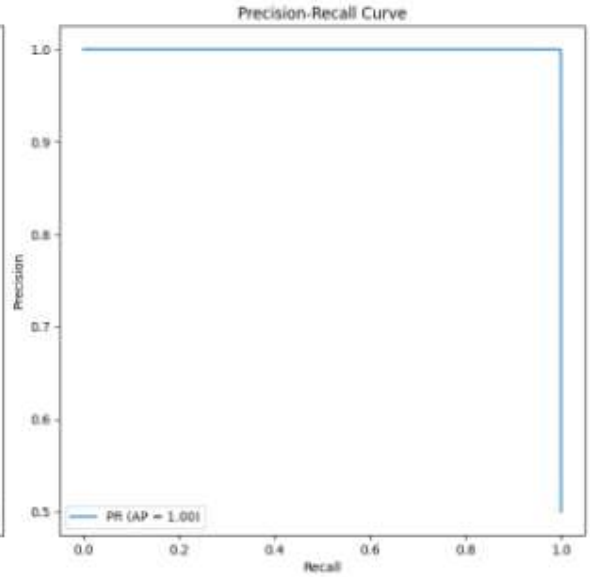


Figure 18 Precision-Recall Curve

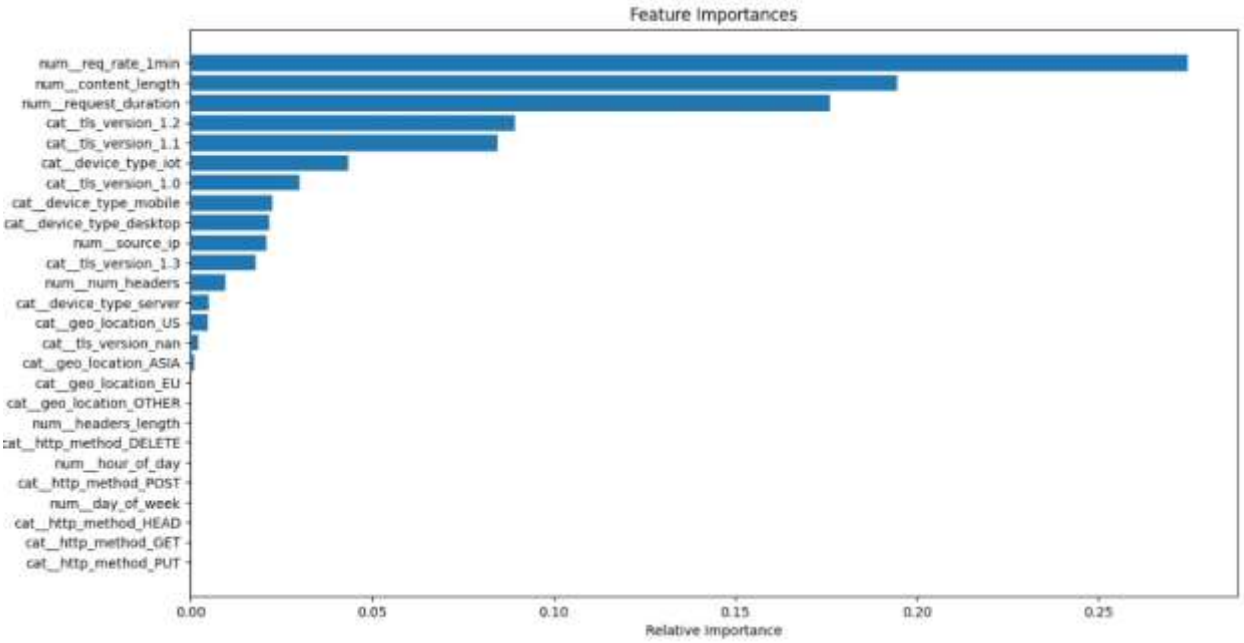


Figure 4.19 Feature graph

4.6.6.2 Model Evaluation for Random forest

Table 4.8 Classification report for model evaluation for Random forest

Class		Precisson	Recall	f1-score	Support
Normal	0	1.00	1.00	1.00	100000
DDoS attack	1	1.00	1.00	1.00	100000
Accuracy				1.00	200000

Table 4.8 provided the classification details of random forest model with the acurray of 100% on the evaluation of image test set; Random forest classifier is based on ensemble learning from multiple decision trees. Figure 4.20 presented the result of confusion matrix which captured all the details in the testing data.

Contingency table/confusion matrix for SVM

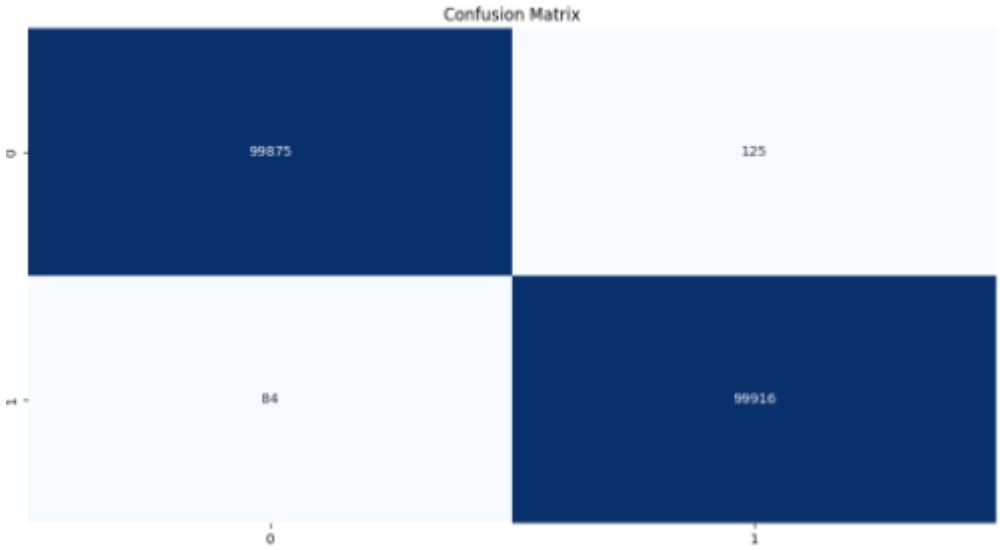


Figure 4.20 Contingency table/confusion matrix for Random Forest

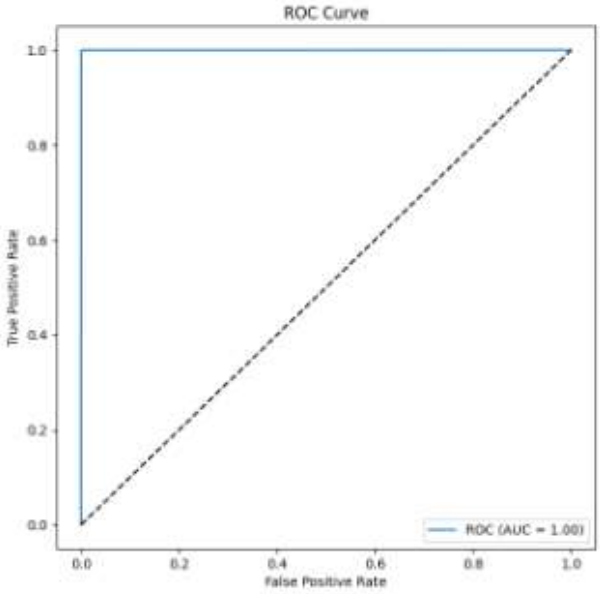


Figure 4.3 ROC curve

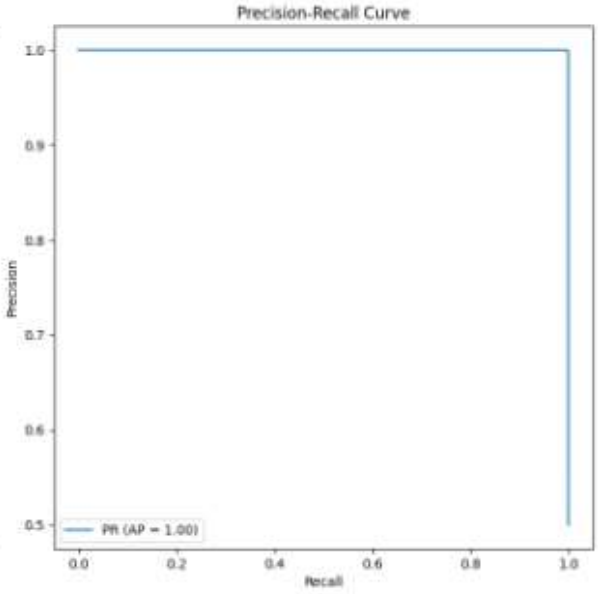


Figure 4.4 Precision and Recall curve

Table 4.9 Comparison Performance of the two models

Evaluation on Test data		
Model	Support	Accuracy
CNN	100000	100%
Random Forest	100000	100%

Table 4.9 presented the comparison performance of the two models. The work achieved the desired results in the comparison table concerning accuracy by the algorithms with 100% in CNN and 100% in Random forest respectively.

4.6.7 Training

This is a very important aspect of system implementation. It enables the users to operate the new system correctly and enjoy its features and resources. This system is user-friendly and requires less training to be used like the other user mobile applications and web applications. Direct training, user handbook or user guide will be used as aids in training the users.

1. Conclusion

Within the landscape of network security, DDoS attacks have emerged as a prominent adversary, especially within expansive networks. The ubiquity of DDoS attacks in large networks has propelled us on a mission to harness the power of machine learning to predict and subsequently thwart these malevolent activities. By delving into the machine learning model's intricacies, we've gained predictive insights into potential DDoS attacks. With the knowledge of pertinent features, we're embarking on a journey to not just predict but actively prevent these attacks from wreaking havoc. This heightened reliance necessitates robust defenses, most notably fortified firewall systems, to strengthen our networks against these relentless attacks. And herein lies the significance of our work: through utilizing deep learning and non-deep learning models, we've culled that both CNN and Random Forest are good models for predicting DDoS attacks. And it's not just about identifying the algorithm but also zeroing in on the key features that are indispensable for prediction.

In this dynamic era, the onus is to keep refining and advancing the machine learning algorithms, equipping them to learn and adapt in tandem with emerging trends and tactics. Ultimately, this journey is fueled by a shared commitment to protect networks and digital spaces from the persistent threat of DDoS attacks. Through predictive insights, presenting the best algorithms, building the best models and an unwavering pursuit of knowledge, we are forging a path towards a safer and more secure digital landscape.

References:

1. Abomhara, M., Khalifa, A., & Hassanien, A. E. (2020). A survey on network anomaly detection techniques: Taxonomy, research challenges, and future directions. *Computer Networks, 178*, 109470. https://doi.org/10.1016/j.comnet.2020.109470
2. Alqahtani, S. A., Wang, Y., & Qiu, M. (2020). A survey of anomaly detection in Internet of Things. *Journal of Network and Computer Applications, 168*, 102742. https://doi.org/10.1016/j.jnca.2020.102742
3. Alzahrani, R., & Alzahrani, A. (2021). Survey of traffic classification solutions in IoT networks. *Computer Networks, 183*, 37–45. https://doi.org/10.1016/j.comnet.2020.107589
4. Anatabine, L., et al. (2022). Deep and reinforcement learning technologies on Internet of Vehicle (IoV) applications: Current issues and future trends. *Journal of Advanced Transportation, 2022*, Article ID 1947886. https://doi.org/10.1155/2022/1947886
5. Bellovin, S. (2000a). Distributed denial of service attacks. Retrieved from http://www.research.att.com/~smb/talks
6. Bellovin, S. (2000b). The ICMP traceback message (Internet Draft). Network Working Group. Retrieved from http://www.research.att.com/~smb/papers/draft-bellovin-itrace-00.txt
7. Cabrera, J. B. D., Lewis, L., & Mehrota, S. (2001). Proactive detection of distributed denial of service attacks using MIB traffic variables—A feasibility study. In *Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management* (pp. 609–622), Seattle, WA.
8. Cheng, X., Chen, Y., Li, X., Wang, H., & Yang, L. T. (2021). A comprehensive survey on cybersecurity using machine learning techniques. *Journal of Network and Computer Applications, 181*, 103026. https://doi.org/10.1016/j.jnca.2021.103026
9. Christos, D., & Aikaterini, M. (2003). DDoS attacks and defense mechanisms: A classification. In *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*. https://doi.org/10.1109/ISSPIT.2003.1341092
10. Dittrich, D., Dietrich, S., & Long, N. (2000). An analysis of the 'Shaft' distributed denial of service tool. Retrieved from http://netsec.gsfc.nasa.gov/~spock/shaft_analysis.txt
11. Gil, T., & Poleto, M. (2001). MULTOPS: A data-structure for bandwidth attack detection. In *10th USENIX Security Symposium*, Washington, DC.
12. Harrison, A. (2000). The denial-of-service aftermath. CNN. Retrieved from http://www.cnn.com/2000/TECH/computing/02/14/dos.aftermath.idg/index.html
13. Huang, Y., & Pullen, J. (2001). Countering denial of service attacks using congestion triggered packet sampling and filtering. In *Proceedings of ICCCN 2001*, Arizona, USA.
14. Huegen, C. A. (2000). The latest in denial of service attacks: Smurfing—Description and information to minimize effects. Retrieved from http://users.quadrunner.com/chuegen/smurf.cgi
15. Kushwah, G., Ranga, V., & Sciences, C. (2021). Distributed denial of service attack detection in cloud computing using hybrid extreme learning machine. *Multimedia Tools and Applications, 80*(29), 1852–1870. https://doi.org/10.1007/s11042-021-10798-2
16. Li, J., Yi, X., & Wei, S. (2020). A study of network security situational awareness in Internet of Things. In *International Wireless Communications and Mobile Computing (IWCMC)* (pp. 1624–1629). https://doi.org/10.1109/IWCMC48107.2020.9148549
17. Martin, B. (2000). Have script, will destroy (Lessons in DoS). Retrieved from http://www.attrition.org
18. McAfee. (2020). Economic impact of cybercrime. Retrieved from https://www.mcafee.com/blogs/other-blogs/mcafee-labs/economic-impact-of-cybercrime/
19. Mohssen, M., Muhammad, B., & Eihab, B. (2016). *Machine learning: Algorithms and applications*. CRC Press. https://doi.org/10.1201/9781315371658
20. Priya, S., Sivaram, S., Yuvaraj, D., & Jayanthiladevi, A. (2020). Machine learning-based DDoS detection. In *Proceedings of the International Conference on Emerging Smart Computing and Informatics (ESCI)*, Pune, India (pp. 234–237). https://doi.org/10.1109/ESCI48226.2020.9167642
21. Somani, G., Gaur, M., Sanghi, D., Conti, M., Rajarajan, M., & Buyya, R. (2017). Combating DDoS attacks in the cloud: Requirements, trends, and future directions. *IEEE Cloud Computing, 4*(1), 22–32. https://doi.org/10.1109/MCC.2017.14

22. Sood, K., Singh, P., Singh, P., & Tyagi, S. (2019). A review of network anomaly detection techniques. *Journal of Network and Computer Applications, 126*, 68–88. https://doi.org/10.1016/j.jnca.2019.03.009
23. Soupionis, Y., & Benoist, T. (2015). Cyber-physical testbed—The impact of cyber attacks and the human factor. In *10th International Conference on Internet Technologies and Secured Transactions (ICITST)*, London, U.K. (pp. 326–331). https://doi.org/10.1109/ICITST.2015.7412114
24. Sultana, N., Chilamkurti, N., Peng, W., & Alhadad, R. (2019). Survey on SDN-based network intrusion detection systems using machine learning approaches. *Peer-to-Peer Networking and Applications, 12*(2), 493–501. https://doi.org/10.1007/s12083-018-0704-3
25. Vishwakarma, R., & Jain, A. (2019). A honeypot with machine learning-based detection framework for defending IoT-based botnet DDoS attacks. In *3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India (pp. 1019–1024). https://doi.org/10.1109/ICOEI.2019.8862720
26. CERT Coordination Center. (1998–2000). CERT Advisories:

- CA-2000-01: Denial-of-service developments. http://www.cert.org/advisories/CA-2000-01.html
- CA-99-17: Denial-of-service tools. http://www.cert.org/advisories/CA-99-17-denial-of-service-tools.html
- CA-98-13: TCP denial-of-service. http://www.cert.org/advisories/CA-98-13-tcp-denial-of-service.html