Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

Detecting DDoS Attacks in Network Traffic Based on Supervised Machine Learning

Mahmood Alfathe
Collage of Information Technology,
Ninevah University,
41002, Mosul, Iraq
Mahmood.alfathe@uoninevah.edu.iq

Aida Mustapha
Faculty of Computing,
Universiti Malaysia
Pahang Al-Sultan
Abdullah,
Pekan 26600,
Pahang, Malaysia
aidam@umpsa.edu.my

Huthaifa L. Mohamed
College of Information
Technology,
Ninevah University,
41002, Mosul, Iraq
Huthaifa.seidan@uoninevah.edu.iq

Salama A Mostafa
Department of Artificial
Intelligence, College of Engineering,
Alnoor University,
Nineveh,41012, Iraq
salama.adrees@alnoor.edu.iq

Yousif Khalid Yousif
College for Computer and
AI, Northern Technical
University, Mosul,
41000, Ninevah, Iraq.
yousif.k.yousif@ntu.edu.iq

Ali H. Al-Shakarchi
College of Information
Technology,
Ninevah University, 41002,
Mosul, Iraq
ali.al-shakarchi@uoninevah.edu.iq

Received: 15 / 10/ 2024 Accepted: 20 / 11/ 2024
Published: 17 / 12 / 2024

Abstract.

One of the major concerns in network security that pose a big challenge to safeguarding networks is distributed denial-of-service (DDoS) attacks. Such attacks often lead to breaches of trust in online systems, cause significant losses in financial markets, and deny services to legitimate users. This study aims to propose a robust method for detecting DDOS attacks accurately.

To accomplish this goal, the study investigated several machine learning algorithms in detecting such attacks utilizing the CIC-DDOS-2019 dataset, a well-known benchmark dataset characterized by its comprehensive coverage of DDOS attacks. Five machine learning algorithms have been evaluated: Random Forest (RF), Naive Bayes (NB), Logistic Regression (LR), J48 Decision Tree, and XGBoost based on their performance in detecting and discriminating between DDoS attacks and benign records. The results show high detection capability, with accuracy rates above 99% for all models except for NB. The RF, LR, J48, and XGBoost algorithms can recognize intricate DDoS assault patterns. In addition to comparing several machine learning methods for DDoS detection, this study provides insight into how these models can be helpful in real-world scenarios for improving network security.

Keywords: Distributed Denial-of-Service (DDoS) attacks, Network Security, Machine Learning, CIC-DDOS-2019 dataset.

1. Introduction

The rapid advancements in digital networks have brought about a significant digital revolution in modern communication and commerce. However, this progress has also exposed these systems to increasing cybersecurity threats, including Distributed Denial of Service (DDoS) attacks, which are among the most severe challenges as they disrupt the services provided by targeted systems. These attacks have become more sophisticated over time, rendering traditional detection methods ineffective. Consequently, there is a growing need to adopt innovative solutions, leveraging modern technologies and artificial intelligence. Artificial Intelligence has played a crucial role in pattern recognition, adapting to emerging threats, and detecting DDoS attacks in real-time [1].

The first DDoS threats emerged in the 1990s, targeting individual systems. A significant turning point occurred in the 2000s with the advent of botnets, enabling attackers to expand their reach and compromise multiple devices [2]. The rise of Internet of Things (IoT) devices has further exacerbated the issue, increasing the complexity of such attacks and empowering cybercriminals [3]. Traditional detection systems have proven ineffective against modern DDoS attacks, relying on outdated techniques incompatible with advanced attack methodologies[4].

A distributed denial of service attack is a cyberattack in which multiple compromised devices are used to overload a target system, server, or network with a stream of traffic. This leads to the inability of the object to respond to legitimate requests, which makes it inaccessible to intended users. Hacked devices, which are often part of a botnet controlled by attackers, simultaneously send many data packets, exploiting vulnerabilities in the network or server. DDoS attacks target various sectors, including finance, healthcare, and e-commerce, causing significant operational and financial disruptions [5].

Figure 1 shows the DDoS attack process, in which attackers use bots to load the victim's network with excessive traffic. Legitimate users trying to access the victim's services face delays or complete unavailability because the system is overloaded. Modern DDoS attacks are characterized by a high degree of complexity. They use advanced tools and methodologies, which makes traditional counteraction strategies inadequate. As a result, organizations are increasingly relying on machine learning and artificial intelligence-based solutions to detect and respond to such attacks dynamically. These systems analyze the structure of network traffic in real-time to distinguish legitimate requests from malicious actions [6].

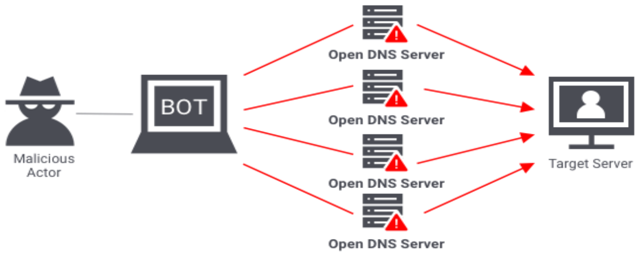


Fig. 1. Distributed Denial of Service attack (DDoS attack) [7]

Machine learning (ML) techniques have introduced advanced approaches for detecting anomalies in typical network behavior by dynamically analyzing network traffic, addressing numerous limitations of older methods. Algorithms like Random Forest, Support Vector Machines (SVM), and Gradient Boosting have demonstrated promise in improving detection rates and mitigating attacks[8].

The primary motivation behind this study is the critical need to strengthen cybersecurity defenses, particularly as DDoS attacks continue to evolve and pose significant threats to sectors such as healthcare and finance. This research aims to enhance the accuracy of detection models, reduce computational complexity, and improve response times for ML algorithms used in identifying DDoS attacks. The study employs the CIC-DDoS2019 dataset to train these algorithms, providing a practical framework applicable to real-world scenarios.

2. Related Work

Advances in digital and information technology, as well as the emergence of machine learning and artificial intelligence, have led to the development of various techniques and solutions tailored to network environments, resulting in significant progress in detecting Distributed Denial-of-Service (DDoS) attacks.

Bhati et al. [9] Propose a new working model utilizing artificial intelligence techniques to achieve the highest accuracy in detecting attacks and intrusions. The model employs three AI techniques: AdaBoost Classifier, Random Forest Classifier, and Logistic Regression. Experiments were conducted on the KDD Cup 99 dataset to identify attacks and detect intrusions. The efficiency and accuracy of this system were demonstrated with an accuracy rate of 99.86% across all categories (Normal, Probe, DoS, U2R, and R2L).

To enhance network security, Hnamte et al.[10] presented a dynamic perspective on Software-Defined Network (SDN) environments. This paper proposed advanced measures to fortify digital infrastructure against intrusion techniques and sophisticated attacks. Three types of datasets were utilized: InSDN, CICIDS2018, and Kaggle DDoS datasets, achieving detection accuracy rates of 99.98%, 100%, and 99.99%, respectively. Additionally, the study provided practical insights into the challenges associated with real-world SDN networks.

Kumari and Pooja [11] proposed an approach based on feature selection techniques to reduce dimensionality and intrusion detection time without compromising accuracy. The approach utilized dimensionality reduction techniques such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Factor Analysis, and Recursive Feature Elimination with Cross-Validation (RFECV). For classifying malicious traffic, machine learning tools were employed, including feature selection techniques, Gaussian Naive Bayes (GNB), Decision Trees (DT), Random Forest (RF), AdaBoost, and Logistic Regression (LR). The study achieved an enhanced accuracy of 99.98% within 0.582 seconds, representing the detection delay time when combining

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

Gaussian Naive Bayes (GNB) with Linear Discriminant Analysis (LDA).

For Internet of Things (IoT) networks, Odumuyiwa et al. [12] trained two clustering algorithms and two deep learning algorithms independently to mitigate DoS attacks. The focus was on Transmission Control Protocol (TCP) attacks and UDP delay attacks. The datasets used included Mirai, Bashlite, and CICDoS 2019. The performance of the four algorithms was evaluated using the Adjusted Mutual Information (AMI) score and accuracy score. Their results demonstrated that the autoencoder performed the best overall.

Najar et al. [13] propose a (BRS + CNN) approach utilizing Balanced Random Sampling (BRS) and Convolutional Neural Networks (CNN) to detect DoS attacks in SDN environments. Various mitigation techniques were employed to prevent spoofed IPs, such as filtering, rate limiting, and iptables rules. Additionally, a monitoring system was proposed that uses rate identification to supervise blocked IP addresses, ensuring efficient handling of legitimate traffic. The proposed system achieved over 99.99% accuracy for binary classification and 98.64% for multi-class classification. Furthermore, it sends detailed contextual information to a designated email address. The efficiency and effectiveness of the proposed DoS mitigation system were evaluated through a series of experiments conducted across three scenarios: attack-free, attack without mitigation, and attack with mitigation.

Finally, Batchu et al. [14] proposed a framework that was implemented using a three-stage deep learning approach: data preprocessing, data balancing, and classification. The data was prepared for further processing during the preprocessing stage. The preprocessed data was then balanced using the Conditional

Generative Adversarial Network (CGAN) to reduce bias toward majority classes. Finally, traffic was classified as either malicious or benign using a Stacked Sparse Denoising Autoencoder (SSDAE) combined with the Firefly-Black Widow Optimization (FA-BWO) hybrid optimization algorithm. All experiments were validated using the CICDDoS 2019 dataset and compared with other techniques. Table 1 summarizes related work and highlights a variety of machine-learning techniques applied to DDoS detection.

Table 1. Summary of the related work

Study	Technique	Application Area	Dataset	Accuracy
Bhati et al. [9]	Ensemble learning approach combining AdaBoost, Random Forest, and Logistic Regression	General network intrusion detection	KDD Cup 99	99.86%
Hnamte et al.[10]	Deep Neural Networks (DNN) for traffic classification	SDN environments	InSDN, CICIDS2018, and Kaggle DDoS	99.98%, 100%, and 99.99%, respectively
Kumari and Pooja [11]	Dimensionality reduction (PCA, LDA, RFECV) combined with machine learning models (GNB, DT, RF, Logistic Regression)	IoT and general intrusion detection	N/A	99.98% accuracy with LDA and GNB in 0.582 s
Odumuyiwa et al. [12]	Machine Learning	(TCP) attacks and UDP delay attacks	Mirai, Bashlite, and CICDoS 2019	N/A
Najar et al. [13]	Convolutional Neural Networks (CNN) combined	SDN environments	N/A	99.99% for binary classification,

	with Balanced Random Sampling (BRS) and iptables rules			98.64% for multi-class classification
Batchu et al. [14]	Three-stage approach: preprocessing, data balancing using Conditional GAN (CGAN), classification with SSDAE and FA-BWO optimization	IoT and SDN environments	CICDoS 2019	N/A

3. Methodology

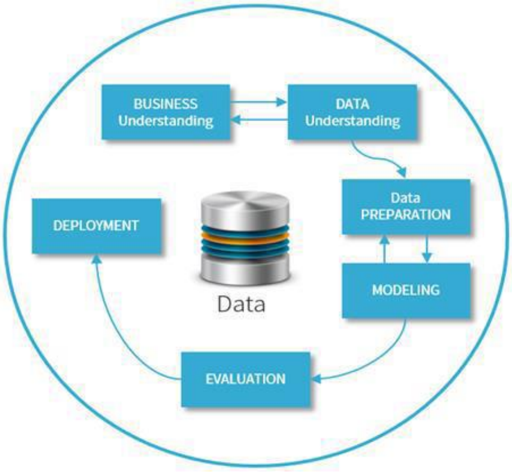
This paper proposed a method for detecting DDOS attacks that arise in networks. DDOS attacks are considered the most severe attacks since they deny services to legitimate users, resulting in a range of consequences, such as financial losses, reputation damage, data vulnerability, etc.

To complete this goal perfectly, the proposed method suggests using machine learning algorithms for the detection of DDOS attacks. After conducting many practical experiments in detecting this type of cyber-attacks, the choice was made on five types of machine learning algorithms that have proven their efficiency and merit in detecting these attacks. In fact, these algorithms were selected from different families, some of them belong to the probabilistic family like (NB) and others depend on the decision tree such as (Random Forest, J48) and some of them relay on statistical methods as (Logistic regression), and for the last algorithm, it was selected from the advanced machine learning

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

families named (XGboost), which is one of the most advanced algorithms that lean on the decision tree.

The objective of the proposed method is to design a lightweight tool that has the capability of detecting DDOS attacks with high accuracy, and low both false negative and positive rates. This is done by making performance comparisons among different learned models (NB, RF, J48, LR, and XGboost) on the selected dataset to select the best one of them. The work has adopted a CIC-



DDOS2019 dataset, a modern, safe benchmark dataset for intrusion detection that mimics the real-world DDOS attack scenarios (PCAPs) created in 2019[15]. Cross-Industry Standard Process for Data Mining (CRISP-DM) is the selected methodology for this work. Figure 2 below illustrates CRISP-DM.

Fig. 2. Illustration of CRISP-DM [16]

Crisp-DM is one of the favorite hierarchical methods in the data mining community. This model is extensively used in data mining processes since it divides the complex data mining task into a set of six simple phases. Such division makes data mining projects easy to execute, manageable, less costly, efficient, and reliable

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

[17]. The following subsections outline the six phases of CRISP-DM as related to our proposed method.

3.1 **Business & Data Understanding**

Business and data understanding concentrate on several key functions: identifying, collecting, and analyzing the selected dataset to fulfill the objectives. Since the proposed work focuses on detecting DDOS attacks, hence, in this correlated phase, the dataset should be acquired from a trusted source. For this work, the CIC-DDOS2019 dataset is obtained from the Canadian Institute for Cybersecurity, which is located at the University of New Brunswick in Fredericton. After determining the selected dataset, the next step includes specifying the dataset quality, such as defining missing values, detecting errors, and reporting any problem encountered when dealing with the dataset. These steps are important to create a comprehensive view of datasets. Note that all attributes should be examined and analyzed in this phase. CICFlowMeter-V3 is adopted to analyze this dataset based on timestamps, port numbers, sources, destination IP addresses, and many other attributes. Table .2 shows the names of features related to the CIC-DDOS2019 dataset.

Table 2. Feature Names in CIC-DOS2019 Dataset

NO	Feature name	NO	Feature name	NO	Feature name	No	Feature name
1	Unnamed: 0	23	Flow Packets/s	45	Bwd Packets/s	67	Bwd Avg Bytes/Bulk
2	Flow ID	24	Flow IAT Mean	46	Min Packet Length	68	Bwd Avg Packets/Bulk

3	Source IP	25	Flow IAT Std	47	Max Packet Length	69	Bwd Avg Bulk Rate
4	Source Port	26	Flow IAT Max	48	Packet Length Mean	70	Subflow Fwd Packets
5	Destination IP	27	Flow IAT Min	49	Packet Length Std	71	Subflow Fwd Bytes
6	Destination Port	28	Fwd IAT Total	50	Packet Length Variance	72	Subflow Bwd Packets
7	Protocol	29	Fwd IAT Mean	51	FIN Flag Count	73	Subflow Bwd Bytes
8	Timestamp	30	Fwd IAT Std	52	SYN Flag Count	74	Init_Win_bytes_forward
9	Flow Duration	31	Fwd IAT Max	53	RST Flag Count	75	Init_Win_bytes_backward
10	Total Fwd Packets	32	Fwd IAT Min	54	PSH Flag Count	76	act_data_pkt_fwd
11	Total Backward Packets	33	Bwd IAT Total	55	ACK Flag Count	77	min_seg_size_forward
12	Total Length of Fwd Packets	34	Bwd IAT Mean	56	URG Flag Count	78	Active Mean
13	Total Length of Bwd Packets	35	Bwd IAT Std	57	CWE Flag Count	79	Active Std
14	Fwd Packet Length Max	36	Bwd IAT Max	58	ECE Flag Count	80	Active Max

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

15	Fwd Packet Length Min	37	Bwd IAT Min	59	Down/Up Ratio	81	Active Min
16	Fwd Packet Length Mean	38	Fwd PSH Flags	60	Average Packet Size	82	Idle Mean
17	Fwd Packet Length Std	39	Bwd PSH Flags	61	Avg Fwd Segment Size	83	Idle Std
18	Bwd Packet Length Max	40	Fwd URG Flags	62	Avg Bwd Segment Size	84	Idle Max
19	Bwd Packet Length Min	41	Bwd URG Flags	63	Fwd Header Length.1	85	Idle Min
20	Bwd Packet Length Mean	42	Fwd Header Length	64	Fwd Avg Bytes/Bulk	86	SimillarHTTP
21	Bwd Packet Length Std	43	Bwd Header Length	65	Fwd Avg Packets/Bulk	87	Inbound
22	Flow Bytes/s	44	Fwd Packets/s	66	Fwd Avg Bulk Rate	88	Label

3.2 Data preparation

Data preparation starts after gaining the desired dataset. This phase is considered as an extensive one since it usually occupies more than 80% of the time needed to complete the project due to the complexity of this step. The key objective of this phase includes identifying, cleaning, and reconstructing the dataset. For our work, the CIC-DDOS2019 is a good choice since it is designed and oriented to evaluate the DDOS attacks in intrusion detection/prevention systems. This dataset contains a wide

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

spectrum of DDOs attacks and benign records which is helpful to provide a real word scenario to evaluate and test Intrusion Detection Systems (IDSs).

CIC-DDOS-2019 dataset includes 50,063,112 records. From these records, 50,006,249 instances related to DDOS attacks, and 56,863 instances are those as representing normal behavior. Each row in this dataset includes 88 attributes that provide rich information related to network traffic. The dataset has 12 different DDOS attacks, like DNS, NetBIOS, NTP, MSSQL, TFTP, SYN, and SNMP, as shown in Table .3 [18]

Table 3. CIC-DDOS-2019 Dataset Attacks

Attack	Counts
Benign	56,863
DNS	5,071,011
LDAP	2,179,930
MSSQL	4,522,492
NetBIOS	4,093,279
NTP	1,202,642
SNMP	5,159,870
SSDP	2,610,611
SYN	1,582,289
TFTP	20,082,580
UDP	3,134,645
UDP-Lag	366,461

Since the selected dataset is considered a big dataset, which contains raw data files of CSV format (11 CSV files), it is difficult to deal with such huge data due to the known limitations in computer resources (processing power, storage space, etc.), as the approximate total size of the data exceeds 17 terabytes, and this size is considered one of the major challenges in dealing with such a volume of data.

The intention was to take a sufficient sample (10% stratified sample) of this data to reflect the total data. First, the CSV files were merged using the panda library in Python to obtain a single file that included all types of DDOS attacks in addition to records of a benign type. The snippet code in Figure 2 below shows the merger operation. The constructed combined CSV file contains all DDOS attacks along with benign records. Next, a stratified 10% sample from the total combined dataset is obtained to ensure fair class distribution. After that, we convert all DDOS attacks type into “ATTACK” labels, reaming the rest records as “BENIGN”. In this way, the proposed tool will be trained on two types of data, attacks and benign, for attack detection.

Cleaning datasets is an important step in the data mining process since it accelerates the processing and minimizes the required memory storage. This step involves handling outlier data like missing, NaN values. Finally, ignoring attributes which have no effect on the detection process. For this reason, Unnamed: 0', 'Flow ID', 'Source IP', 'Destination IP', 'Timestamp', 'SimillarHTTP have been eliminated from the dataset. After applying the previous preliminary preprocessing, the statistics of the data remaining for processing are (1949713) and (5631) instances for attack and benign classes, respectively.

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

3.3 Modeling

In this phase, different machine learning models have been assessed. For this work, various machine-learning algorithms have been selected, as mentioned previously. These algorithms are Navie Bayes (NB), Decision Tree (Random Forest (RF), and J48), Logistic Regression (LR), and XGBoost. A brief overview of each algorithm is provided as follows.

- **Navie Bayse:** This algorithm relies on the Bayesian theorem and is considered an efficient classification algorithm. NB concentrates on the conditional probability of records in the dataset, such that for each instance Xi in training data related to class C, the probability of the class is determined based upon its attributes $X_{1,}$ $X_{2,\ldots}$ X_n. Hence, the class label would be predicted with maximum posterior probability [19]. Bayes's theorem is illustrated in Equation 1 below [20].

  $$P(B) = P(A).P(A)|P(B) \qquad (1)$$

  Where P represents the probability, PAB denotes the posterior probability, P(A) represents the prior probability, and $P(B)$ is the past probability of the predictor.

- **Random Forest:** Leo Breiman from the University of California proposed the RF decision tree [18]. The basic component of RF is many decision trees that are characterized as independent from each other. Voting between these sub trees is used to determine the winning class [19].
- **J48:** Decision tree j48 is a classification algorithm that is considered an extension of the C4.5 tree proposed by Quinlan in 1993. Like all decision trees, this tree relies on the divide and conquer concept. J48 extensively split the dataset based on the attributes to maximize gain. In the J48 tree, each path from the

root node to the leaf node represents a classification rule. The decision tree may not give high accuracy in classification if there are many classes, unlike if the classification process is carried out on only two classes, where the decision tree records the highest accuracy [9]. For this work, the J48 was selected due to its high detection rate [20].

- **Logistic Regression:** Logistic Regression (LR) is considered a supervised algorithm for classification problems. Its principal work relies on the fact that independent features can be utilized to predict dependent features. LR predicts the class probabilities based on the sigmoid function and gets the fitted data through maximize likelihood estimation. In other words, the regression process can estimate the dependent variable, X, by knowing a set of values related to the independent variable, Y. Thus, it tries to find the excellent fitting line that reflects the variable's relation[21].

$$L_n\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k \qquad (2)$$

Where L_n refers to the regression function, p is the variable's probability, X represents the risk factor, and β is constant equal to 1.

- **XGBoost:** Extreme Gradient Boosting (XGBoost) is a powerful, efficient, and scalable algorithm based on gradient boosting concept. Because of its effectiveness and adaptability, it is frequently utilized for both regression and classification problems. This algorithm uses advances like scalable tree construction, efficiently handling missing data, and reducing overfitting. In addition, the XGBoost algorithm could optimize the use of parallel processing, which is very important when

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

dealing with large data sets, and it is considered as faster than other methods executed on a single machine[22].

3.4 Implementation and Evaluation

This section thoroughly explains the implementation of the proposed DDOS attack detection, including the experimental design, methods used, and the work's flowchart. Following this, we show and discuss the findings from the experiments that were carried out.

The experiments were conducted on a machine equipped with an Intel(R) Core (TM) i5-2410M CPU 2.30GHz. This processor facilitated the efficient training and evaluation of the machine learning models. The machine was also configured with 8 GB RAM, a Windows 11 operating system. Python programming language has been adopted along with the Jupyter Notebook as a programming interface; this is accomplished by using the release provided by Anaconda, which makes remarkable integration between Python and Jupyter Notebook, offering an effective environment for creating and testing machine learning models.

The flowchart in Figure .3 depicts the procedure steps for the proposed work and makes it easy to track the implementation of each action step. This flowchart covers important steps, starting from selecting a dataset, data preparation, modeling, and finalizing with assessment and evaluation of the selected models. The steps are described in detail below.

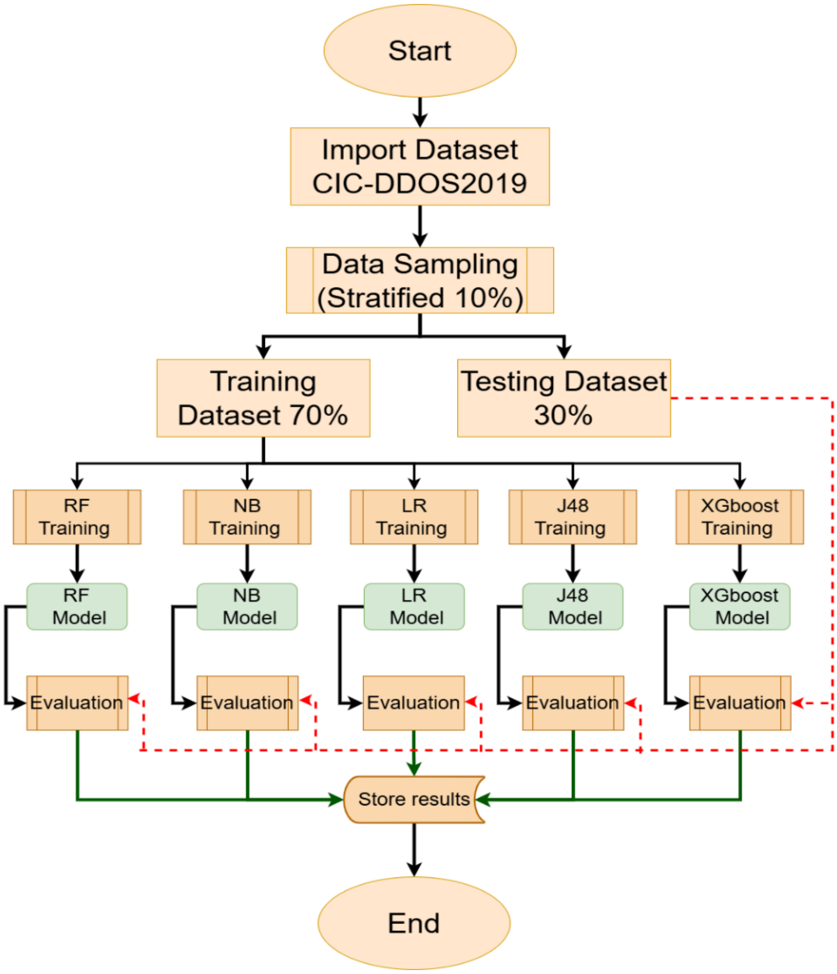


Fig. 3 DDOS Detection and Evaluation Flowchart

First, the data for DDOS detection is imported. This data undergoes a series of initial processing, including merging the data files, which consist of 7 CSV files, where a 10% sample of the total data is taken to form the final data to be used. Then, this data is projected to pre-processing, including dealing with fields with NAN values and empty fields. Secondly, the data set is partitioned into training and testing datasets. The work adopted 30% and 70 % datasets for training and testing, respectively.

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

Next, the training phase is completed, in which five machine learning algorithms are chosen. They are RF, NB, LR, J48, XGBoost. Each learner receives the same training dataset and starts its kernel to produce a learned model. Finally, after the training phase is completed, the evaluation phase is started. In this phase, the performance of each model is determined based on five chosen evaluation metrics (Accuracy, recall, precision, specificity, and F-measure), which test each model based on an unseen testing dataset. All evaluation results will be stored and compared to choose the best model, which is then utilized to deploy the final detection tool.

Several evaluation metrics are utilized to evaluate the efficiency of our proposed tool. These metrics could reflect the performance of discriminating against malicious and benign traffic. These evaluations are derived from the well-known Confusion matrix (CM), which reveals all possible detection cases. Figure 4 illustrates what CM is made up of [22].

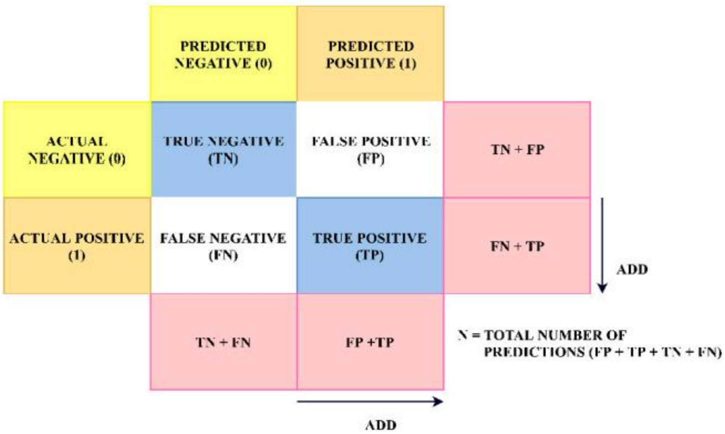


Fig. 4 CM Representation

The following is an explanation of CM components:

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

- TN: is the quantity of benign cases that are accurately categorized.
- FP: is the quantity of benign cases that are misclassified.
- FN: is the quantity of assault cases that were misclassified.
- TP: The number of assault instances that are accurately classified.

Several evaluation metrics that have been adopted are **accuracy**, **precision**, **recall**, specificity**,** and **F-measure** . All these metrics are calculated based on information provided in CM matrix as shown below.

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (3)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (4)$$

$$Precision = \frac{(TP)}{(TP+FP)} \quad (5)$$

$$Specificity = \frac{(\boldsymbol{TN})}{(\boldsymbol{FP}+\boldsymbol{TN})} \quad (6)$$

$$F - measure(F) = \frac{2.R.P}{(R+P)} \quad (7)$$

4. Results

The results of the evaluation metrics of **RF, NB, LR, J48, and XGBoost** are shown in Table 4. Experimental results reveal that both J48 and RF show high performance for all metrics, and the scores recorded are nearly perfect. Nevertheless, XGBoost also shows interesting high accuracy that reaches (0.99985). However, NB has a substantially poorer F-measure (0.041969), precision (0.023109), and recall (0.228242), suggesting that it has trouble with this dataset.

Al-Noor Journal for Information Technology and Cyber Security
December (2024); 1 (0): 23- 48
DOI: https://doi.org/10.69513/jnfit.v1.i0.a2

Table 4. Evaluation Results

	Accuracy	Precision	Recall	Specificity	F-measure
RF	0.999992	0.997343	1.000000	0.999992	0.998670
NB	0.969998	0.023109	0.228242	0.972140	0.041969
LR	0.997686	0.890459	0.223801	0.999921	0.357700
J48	0.999994	0.999554	0.999554	0.999997	0.997433
XGBoost	0.999985	0.994700	1.000000	0.999985	0.997343

On the other hand, LR shows a lower F-measure (0.357700), which is directly impacted by the recall value (0.223801), although LR has interesting accuracy (0.997686) and specificity (0.999921) compared to its F-measure. The low Recall and F1-measure results show that the NB and LR cannot detect the minority class (normal request) compared to other classifiers. From the results, it can be said that both J48 and RF have balanced performance for all metrics, reflecting their robustness. Followed by XGBoost, which shows high recall but low precession. In the opposite of the Naïve Bayse model, which shows a significant decrease in accuracy due to low precession metrics, which is attributed to the misclassification of many instances. Figure 5 shows a model comparison using the Receiver Operator Characteristics (ROC) curve. In ROC, for every classifier, the True Positive Rate (TPR) is plotted against the False Positive Rate (FPR) using a receiver operating characteristic curve.

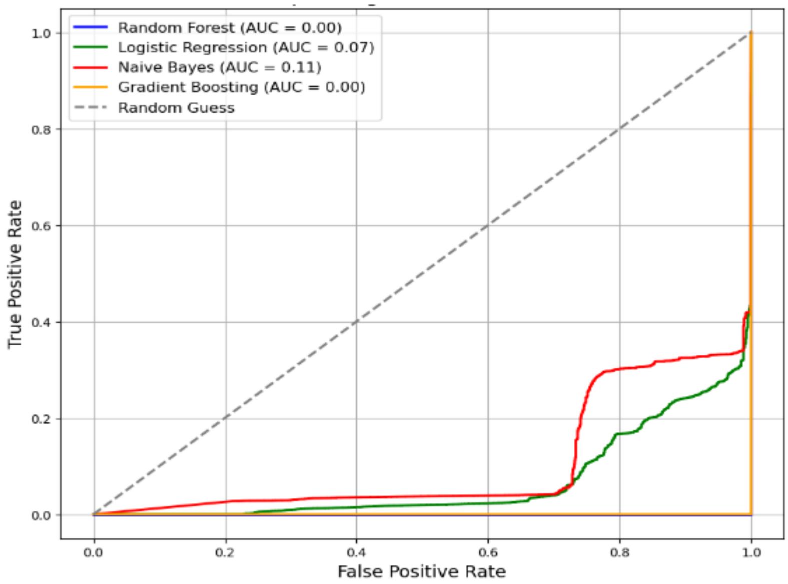


Fig. 5. Receiver Operator Characteristics (ROC) Comparison

This analysis aids in assessing each model's ability to distinguish between the positive and negative classes at different threshold values. The figure shows superior results for both XGBoost and RF achieved in the vertical to-left corner (not visible). Hence, J48 is not plotted since it has the same RF and score.

5. Conclusion

This paper presents a DDOS detection method after extensively assessing the effectiveness of five classifiers named (NB, LR, J48, XGBoost, and RF) trained on the CIC-DDOS-2019 dataset. The aggregated overall results showed that advanced classification methods like RF, XGBoost, and J48 are highly recommended for DDOS detection tasks that exceed 99.99% for accuracy. NB shows poor performance due to a larger number of misclassified instances. In fact, the imbalanced dataset is the main reason for the degradation of all lower metrics. Presently, the intention is to focus on low-performance models and try to enhance their classification accuracy by exploring and finding the impact feature and utilizing feature engineering methods in addition to reconsidering imbalanced datasets and using all modern technologies to deal with such unbalanced datasets.

References

[1] M. Mittal, K. Kumar, and S. Behal, "Deep learning approaches for detecting DDoS attacks: A systematic review," *Soft computing*, vol. 27, no. 18, pp. 13039–13075, 2023.

[2] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, pp. 39–53, 2004.

[3] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[4] H. H. Jazi, H. Gonzalez, N. Stakhanova, and A. A. Ghorbani, "Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling," *Computer Networks*, vol. 121, pp. 25–36, 2017.

[5] Y. Al-Hadhrami and F. K. Hussain, "DDoS attacks in IoT networks: a comprehensive systematic literature review," *World Wide Web*, vol. 24, no. 3, pp. 971–1001, 2021.

[6] A. Singh and B. B. Gupta, "Distributed denial-of-service (DDoS) attacks and defense mechanisms in various web-enabled computing platforms: issues, challenges, and future research directions," *International Journal on Semantic Web*

and Information Systems (IJSWIS), vol. 18, no. 1, pp. 1–43, 2022.

[7] “OneLogin. (n.d.). Diagram of a DDoS attack . Retrieved December 8, 2024, from https://www.onelogin.com.”

[8] K. S. Sahoo *et al.*, “An evolutionary SVM model for DDOS attack detection in software defined networks,” *IEEE access*, vol. 8, pp. 132502–132513, 2020.

[9] N. S. Bhati and M. Khari, “An ensemble model for network intrusion detection using adaboost, random forest and logistic regression,” in *Applications of Artificial Intelligence and Machine Learning: Select Proceedings of ICAAAIML 2021*, Springer, 2022, pp. 777–789.

[10] V. Hnamte, A. A. Najar, H. Nhung-Nguyen, J. Hussain, and M. N. Sugali, “DDoS attack detection and mitigation using deep neural network in SDN environment,” *Computers & Security*, vol. 138, p. 103661, 2024.

[11] P. Kumari and A. K. Jain, “Timely detection of DDoS attacks in IoT with dimensionality reduction,” *Cluster Computing*, pp. 1–19, 2024.

[12] V. Odumuyiwa and R. Alabi, “DDOS detection on internet of things using unsupervised algorithms,” *Journal of Cyber Security and Mobility*, pp. 569–592, 2021.

[13] A. A. Najar and S. M. Naik, “Cyber-secure SDN: A CNN-based approach for efficient detection and mitigation of DDoS attacks,” *Computers & Security*, vol. 139, p. 103716, 2024.

[14] R. K. Batchu, T. Bikku, S. Thota, H. Seetha, and A. A. Ayoade, “A novel optimization-driven deep learning

framework for the detection of DDoS attacks," *Scientific Reports*, vol. 14, no. 1, Art. no. 1, 2024.

[15] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *2019 international carnahan conference on security technology (ICCST)*, IEEE, 2019, pp. 1–8.

[16] A. H. Al-Shakarchi, S. A. Mostafa, M. Z. Saringat, D. A. Mohammed, S. H. Khaleefah, and M. M. Jaber, "A Data Mining Approach for Analysis of Telco Customer Churn," in *2023 Al-Sadiq International Conference on Communication and Information Technology (AICCIT)*, 2023, pp. 23–27. doi: 10.1109/AICCIT57614.2023.10218161.

[17] C. Schröer, F. Kruse, and J. M. Gómez, "A Systematic Literature Review on Applying CRISP-DM Process Model," *Procedia Computer Science*, vol. 181, pp. 526–534, 2021, doi: https://doi.org/10.1016/j.procs.2021.01.199.

[18] M. C. P. Saheb, M. S. Yadav, S. Babu, J. J. Pujari, and J. B. Maddala, "A Review of DDoS Evaluation Dataset: CICDDoS2019 Dataset," in *Energy Systems, Drives and Automations*, J. R. Szymanski, C. K. Chanda, P. K. Mondal, and K. A. Khan, Eds., Singapore: Springer Nature Singapore, 2023, pp. 389–397.

[19] N. H. Al-A'araji, S. O. Al-Mamory, and A. H. Al-Shakarchi, "Constructing decision rules from naive bayes model for robust and low complexity classification," *International Journal of Advances in Intelligent Informatics*, vol. 7, no. 1, pp. 76–88, 2021.

[20] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine learning*, vol. 29, pp. 131–163, 1997.

[21] H. Jain, A. Khunteta, and S. Srivastava, "Churn prediction in telecommunication using logistic regression and logit boost," *Procedia Computer Science*, vol. 167, pp. 101–112, 2020.

[22] S. S. Dhaliwal, A.-A. Nahid, and R. Abbas, "Effective intrusion detection system using XGBoost," *Information*, vol. 9, no. 7, p. 149, 2018.