Computers, Materials & Continua
DOI: 10.32604/cmc.2022.025048
Article

Tech Science Press

Machine Learning with Dimensionality Reduction for DDoS Attack Detection

Shaveta Gupta[1], Dinesh Grover[2], Ahmad Ali AlZubi[3,*], Nimit Sachdeva[4], Mirza Waqar Baig[5] and Jimmy Singla[6]

[1]IK Gujral Punjab Technical University, Jalandhar, 144603, India
[2]Department. of Electrical Engineering and Computer Science, Punjab Agriculture University, Ludhiana, 141004, India
[3]Department of Computer Science, Community College, King Saud University, Riyadh, 11437, Saudi Arabia
[4]Vunsol Private Limited, Mohali, 160055, India
[5]Department of Electrical Engineering, FAST National University, CFD Campus, Faisalabad, 44000, Pakistan
[6]School of Computer Science and Engineering, Lovely Professional University, Punjab, 144001, India
*Corresponding Author: Ahmad Ali AlZubi. Email: aalzubi@ksu.edu.sa
Received: 09 November 2021; Accepted: 29 December 2021

Abstract: With the advancement of internet, there is also a rise in cybercrimes and digital attacks. DDoS (Distributed Denial of Service) attack is the most dominant weapon to breach the vulnerabilities of internet and pose a significant threat in the digital environment. These cyber-attacks are generated deliberately and consciously by the hacker to overwhelm the target with heavy traffic that genuine users are unable to use the target resources. As a result, targeted services are inaccessible by the legitimate user. To prevent these attacks, researchers are making use of advanced Machine Learning classifiers which can accurately detect the DDoS attacks. However, the challenge in using these techniques is the limitations on capacity for the volume of data and the required processing time. In this research work, we propose the framework of reducing the dimensions of the data by selecting the most important features which contribute to the predictive accuracy. We show that the 'lite' model trained on reduced dataset not only saves the computational power, but also improves the predictive performance. We show that dimensionality reduction can improve both effectiveness (recall) and efficiency (precision) of the model as compared to the model trained on 'full' dataset.

Keywords: DDoS (Distributed denial of service); internet; ML (machine learning); accuracy

1 Introduction

In all realms of business and industry, including banking, social media, e-mail, and university e-Services, network security has been crucial [1]. Attacks have been launched against a variety of web and network services. The DDoS attack is the supreme culprit to exploit the limitations of the internet [2]. When a popular website is not up, or the customers are deprived of access to a site, often the

primary reason is a DDoS attack. The rationale behind facilitating the denial-of-service attack is to overload the victim with traffic, often more than its capacity, because of which the server becomes inoperable as shown in Fig. 1. Hackers are constantly developing new types of Distributed Denial of Service (DDoS) attacks that target both the application and network layers.

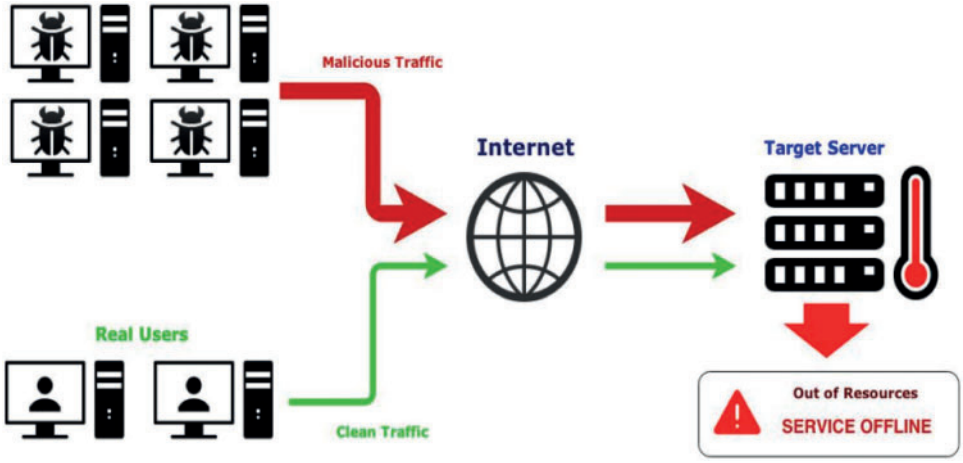

Figure 1: Typical Distributed Denial of Service (DDoS) attack

In the last two decades, DDoS attacks are increasing at an alarming rate both in frequency and severity. In February 2020, it was detected on Amazon Web Services [3]. It was 2.3 Tbsp. This attack is caused for three days of "Elevated Threats" for amazon's shield staff. Similarly, it also happened on GitHub, which is an online code monitoring system utilized by several millions of developers in 2018. DDoS attacks can be driven by a variety of factors, including friendly competition, hacktivism, and acts of vengeance [4]. There are vast weaknesses present in the network architecture that attracts hackers and intruders to launch DDoS attacks. Some of the internet characteristics that invite hackers to launch DDoS attacks are the deterministic nature of Internet Protocols, the stateless nature of routers, Lack of Authenticity on the internet, etc. As DDoS attacks are growing exponentially, and it is a big threat in the present digital world, so researchers have developed a variety of solutions to cope with them. Some attackers, on the other hand, are clever enough to get beyond these defenses [5].

Whenever the attacker attacks on some website or a server, it's important to filter the attack flow from the usual flow so that the genuine users need not suffer. Attack detection methodologies do the same thing i.e., filter out the legitimate traffic from attack traffic. A prerequisite for attack detection is to gather enough information about the network traffic to analyze it for proper filtration. Broadly, it is categorized into two types [6] as shown in Fig. 2:

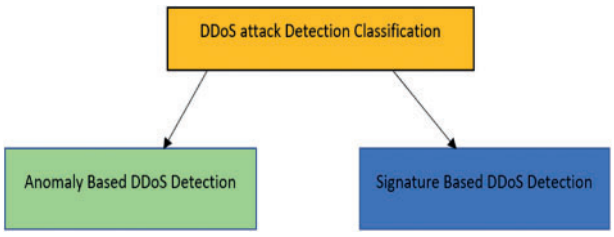

Figure 2: DDoS attack detection methodologies

Signature Based DDoS detection (Misuse Detection): In these methods, a list of known signatures of attacks needs to be stored in the database and then traffic is monitored based on these signatures. If a match occurs, then it generates an alarm of suspicious traffic. Its biggest benefit is that it mostly gives 100% accurate results; however, its major disadvantage is its inability to detect unknown attacks [7–10].

Anomaly-Based DDoS detection: In these methods, the system monitors the traffic data against a database containing features of normal data and any deviation from these features generates an alarm. Under this research work, we have used an anomaly-based detection methodology [7–10].

Existing attack detection approaches [11–15] aim to detect ongoing DDoS attacks. Characterization of DDoS attacks helps to discriminate DDoS attacks from genuine users. However, no foolproof solution has yet been discovered. The researcher aims to lessen the false positive and false negative rates of detection but making them zero is impossible [16,17].

Different researchers used different methodologies to detect DDoS attacks like statistical technique [18], neural network [19], fuzzy logic [20], or machine learning [21]. Machine Learning is a data processing technique for creating analytical models that is automated. It is a subset of artificial intelligence predicated on the idea that computers can learn from data, recognize trends, and make judgments with little human intervention. Demand and importance of machine learning are increasing day by day among scientists, data analysts, and the corporate world. The difference between machine learning and statistical methods is their purpose. Statistical methods work best when we need to infer something from the data set. Machine learning, on the other hand, works effectively when the goal is to make predictions based on a set of data. As a result, machine learning is an algorithm that can learn from data without the need for specific laws, as is the case for conventional computer programs.

The machine learning mechanisms are also widely used to detect attacks on the networks in centralized environments, such as cloud computing, software-defined networks, etc. However, data is usually the only requirement for machine learning.

That's why this research work is going to use a data set [22] that has sufficient rows to train the machine learning model. This cleaned dataset contains 60 features to train the machine learning model. However, according to the "curse of dimensionality," the more features in a data set, the greater the risk of the model overfitting the data [23]. Since overfitted models cannot be generalized well to out-of-time data, so the next step is to evaluate if there is a way to reduce the input feature list without compromising the performance of the system. Furthermore, for the robustness and trustworthiness of the models, practitioners also need insights on which features contribute to predictive accuracy and they should be interpretable. Therefore, we select the best features as picked by the algorithm and reduce our dataset by using only those features. Our contributions are summarized as follows:

- The system undergoes a series of steps to pre-process the dataset.
- On the 'full' data set, various machine learning models are implemented. Based on performance measures, a comparative analysis of various machine learning models was conducted.
- Then, using 'feature importance' and 'Shapley value,' we used dimensionality reduction to the data set, picking just the highest performing features in our data.
- Finally, Random Forest algorithm is applied on the reduced dataset, and the performance is compared with the best performing model using the full dataset.

2 Machine Learning Models

Machine Learning is a data processing technique for creating analytical models that is automated. It is divided into three types: supervised, unsupervised, and reinforcement learning. We will employ supervised machine learning techniques in this study, which are briefly outlined below [24]:

2.1 Logistic Regression

It is the most basic and widely used machine learning algorithm for two-class classification problems. It is a statistical method to predict binary classes. Linear Regression assumes that the data follows a linear function and gives continuous output whereas the Logistic Regression model the data using Sigmoid Function and gives constant output. The sigmoid function, also known as the logistic function, generates an S-shaped curve that may transfer any real-valued number to a value between 0 and 1.

2.2 Decision Tree

It is a supervised learning technique that can be used to solve problems like classification and regression. Internal nodes carry dataset attributes, branches represent decision rules, and each leaf node represents the conclusion in a tree structured classifier.

2.3 KNN (k-Nearest Neighbors)

It is an easy approach to sort the data as shown in Fig. 3.

- Start with a dataset with identified categories.
- Then add a new set of rows of data set that we need to classify.
- Then categorize the new cell by studying the nearest annotated cells.
- If $k = 1$, the algorithm will look for a neighbor who is closest to a new cell. If $k = 11$, the 11 closest neighbors would be used.

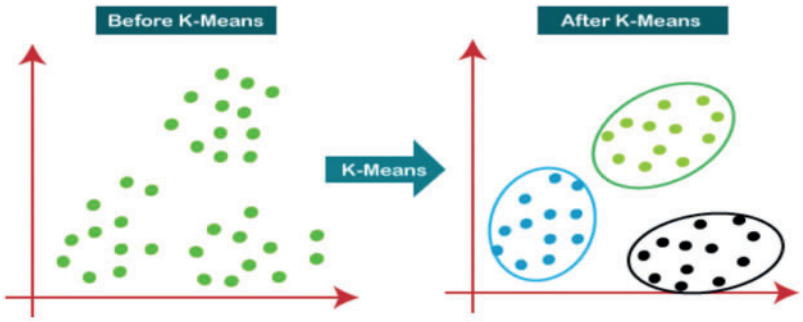

Figure 3: KNN model

2.4 Random Forest Machine Learning Model

A random forest is a supervised machine learning system that uses decision tree algorithms to build it. This algorithm is used to anticipate behavior and outcomes in a variety of industries, including banking and e-commerce. Small changes to the training set might result in drastically different tree architectures, which is why decision trees are so sensitive to the data they're trained on. Random forest

takes use of this by enabling each tree to sample from the dataset at random with replacement, resulting in unique trees Fig. 4. Bagging is the term for this procedure.

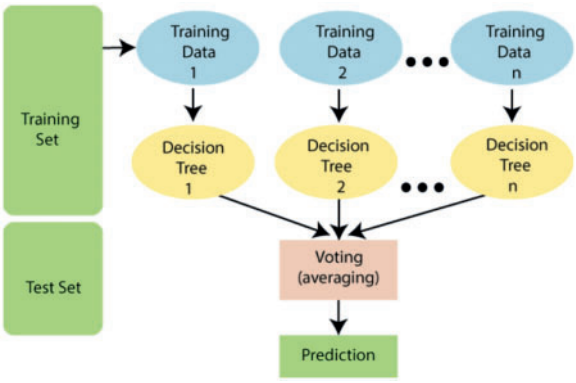

Figure 4: Random forest model

Step 1: Pick K data points from the training set at random
Step 2: For the data points you've picked, make decision trees (Subsets).
Step 3: Choose a N for the number of decision trees you want to make.
Step 4: Go oversteps 1 and 2 again.
Step 5: Locate each decision tree's projections for new data points and assign them to the category with the most votes.

2.5 Support Vector Machine (SVM)

In this algorithm, for the classification of data points, the system will find a hyperplane in N-dimensional space that can do it. There can be a vast hyperplane that can do this job, but algorithm needs to choose that who has the maximum margin (Maximum distance between data points for both classes) as shown in Fig. 5.

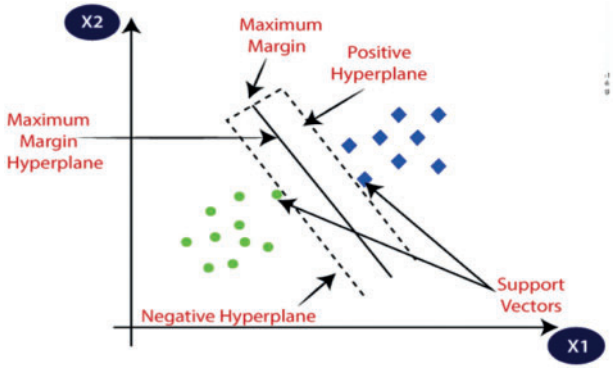

Figure 5: SVM model

2.6 NBC (Naive Bayes Classifier)

It is a probabilistic machine learning system that's commonly used to classify data sets. The Bayes Theorem is used to support this. Its advantages are that they give us fast results and are easy to

implement. But its major disadvantage is that this algorithm demands predictors to be independent, but in most of the real scenario's predictors are dependent.

3 Related Work

This section contains an overview of several publications on the machine learning approaches used to detect DDoS attacks:

Prasad et al. [22] provided a DDoS detection method using machine learning and Stochastic Gradient Boosting. DDoS attacks are detected using machine learning in a non-linear way. Different Classifiers are used for intrusion detection. XGBOOST is a program that implements an algorithm. For testing and training, a 2:1 data set ratio is used.

Pérez-Díaz et al. [25] demonstrated a modular and supported framework for detecting and mitigating the LR-DDoS attacks in SDN (Software Defined Networking) settings. The Intrusion Detection System was trained using the six machine learning algorithms. The authors use ML techniques like SVM, Random Forest and J48The accuracy of these models was also evaluated using the DoS dataset from the Canadian Institute of Cybersecurity. According to the data, the suggested solution achieved a detection rate of 95%.

Karan et al. [26] presented a detection model for detecting DDoS attacks in an SDN environment. In this proposed model, two layers of protection are used. The evolved framework initially detects attacks based on signatures. These attacks are detected using Snort. Following that, two classifiers from machine learning techniques were used to construct a qualified model. These classifiers help vector machines and deep neural networks. This is followed by a comparison of the two classifiers. As a result, the model's accuracy is 74.3%, and the DNN model is more efficient (with an accuracy of 84.3%).

Nanda et al. [27] used machine learning algorithms to build a model. The model was trained by using information gleaned from previous attacks or interactions to recognized malicious attacks and contacts. To suggest the model, the most used ML techniques are Decision Table, Naïve Bayes, Bayesian Network and, C4.5. This model describes the network that has been. After comparing the results, the accuracy of the Bayesian Network was found to be higher than that of the other models, at 91.68 percent.

Silveira et al. [28] introduced a smart detecting gadget. This gadget aids in the detection of network DoS or DDoS attacks. The researchers employed the Random Forest Tree Algorithm, a machine learning technique, to develop this model, which classifies network traffic depending on the samples provided during the training phase. A series of tests are often performed to evaluate the performance of this scheme. As a result of these investigations, the given method is more realistic and has improved efficiency when compared to the most recent current system available in the literature on this subject.

Li et al. [29] defined a method that uses deep learning to identify DDoS attacks on a network. The suggested model will achieve the outcome by using the network's background of traffic dynamics as well as other network attack operations. The findings of this study also showed that the deep learning method is more reliable, effective, and effective.

Elsayed et al. [30] extensively examined the different ML methodologies used by multiple researchers to detect DDoS attacks in the SDN environment. This study looked at the specific shortcomings that have been found in conventional models. Per technique has been tested in accordance with different performance criteria. In this job, four techniques are compared: SVM, Random Forest, and Naïve Bayes and J48. It is discovered that the J48 machine learning algorithm is the best method

for detecting DDoS attacks in an SDN environment since it is more accurate than other current approaches.

4 Proposed Algorithm

This section described the proposed algorithm to detect DDoS attacks. This research work is going to present a paradigm for decreasing data dimensions by identifying the most significant features that influence forecast accuracy. This shows that training a 'lite' model on a smaller dataset not only saves time and effort, but also enhances predictive performance.

This research work is going to introduce an algorithm that can detect the DDoS attacks as described in Fig. 6.

1. First, collect the data.
2. The data has been cleaned.
3. Select 10% of data at random, i.e., 1048576 Flows.
4. Apply Dimensionality Reduction on the cleaned data set.
5. Then the data set is split up into two parts. In this research work, 60% of the subset data is used to train Random Forest machine learning model.
6. Second, the trained model is tested on the remaining 40% of the data subset.
7. Performance Evaluation of trained model has been done based on metric values.

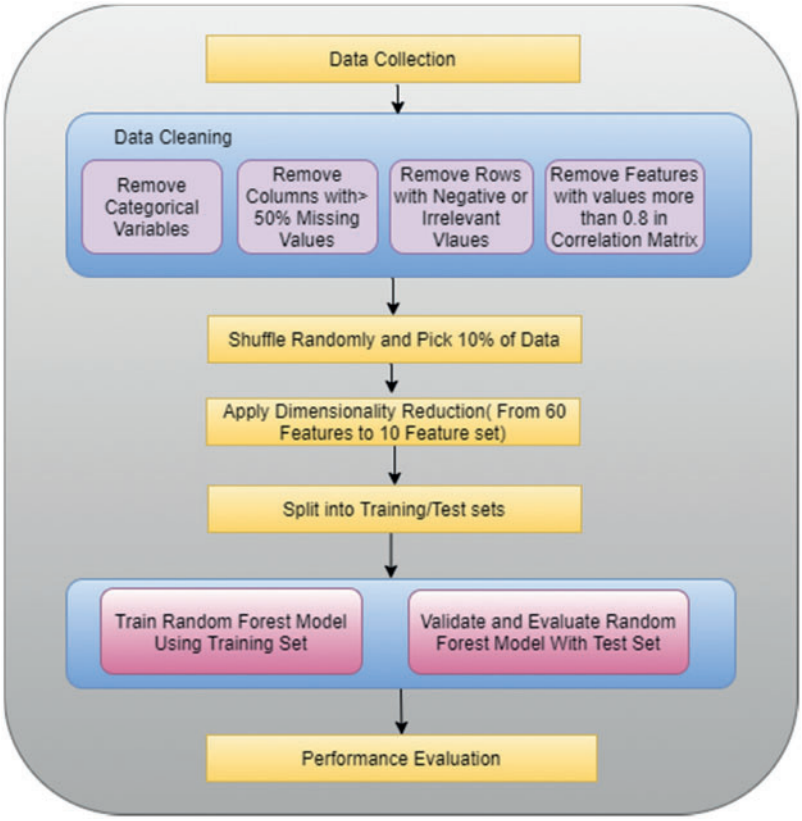

Figure 6: Proposed algorithm

4.1 Data Set and its Processing

For this research, a data set was collected from three open data sets that had already been done [22] and are listed below in Tab. 1.

Table 1: Description of dataset

Data set	File name	Tools/attack type	File size
CSE-CIC-IDS2018-AWS	Friday 16/2/2018 (all-pcaps)	DoS-SlowHTTPTest DoS-Hulk	47
	Thursday15/2/2018 (all-pcaps)	DoS-GoldenEye DoS-Slowloris	46
	Wednesday21/2/2018 (all-pcaps)	DDoS-LOIC-UDP DDoS-HOIC	76
	Tuesday 20/2/2018	DDoS attacks-LOIC-HTTP	52
	Tuesday 20/2/2018 Traffic for ML_CICFLOWMETER.csv	DDoS-LOIC-UDP	
CICIDS2017	Monday-Working Hours.pcap	Benign traffic	10.8
	Friday-Working Hours.pcap	DDoS-LOIC Port Scan	8.8
CIC DoS Dataset (2016)	AppDDoS.pcap	Slowbody2 Ddosim Goldeneye slowheaders Hulk slowloris rudy slowread	4.6

Tab. 2. Mentioned that the total number of flows initially in the data set is 1294529(Imbalanced Flows). To make the data set balanced, so that machine learning model can be trained effectively, we have removed approx. 6000000 flows from Label DDoS. Finally, in the balanced dataset has12794627 flows. So, it is computationally very cumbersome to incorporate with approximately 12 M rows approximately, we pre-process the data to come up with a significant number of rows to optimize the results. Fig. 7. Shows a graphical representation of a balanced data set.

Table 2: Flow details in balanced and imbalanced data sets

Data set	Balanced	Imbalanced
Label: DDoS	6472647	1294529
Label: Benign	6321980	6321980
Total Flows	12794627	7616509

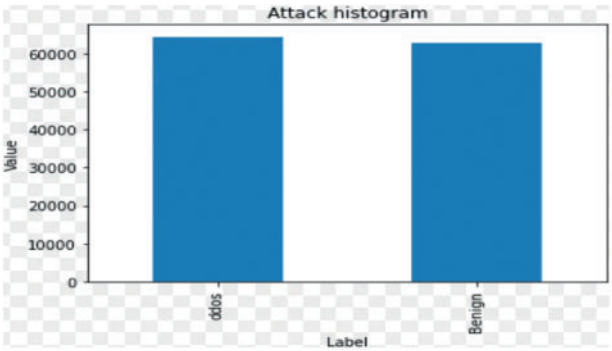

Figure 7: Graphically representation of a balanced dataset

We perform a series of steps to process the dataset to get a subset that is enough to apply the proposed algorithm as shown in Fig. 8.

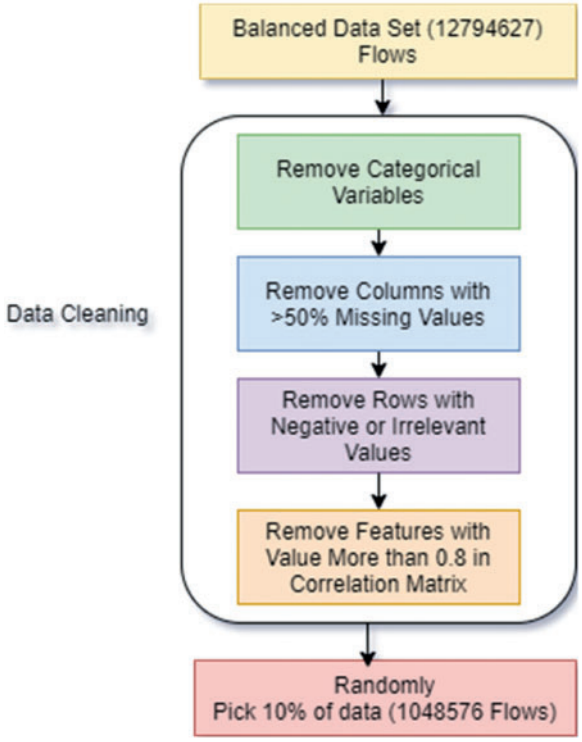

Figure 8: Dataset processing

1. The original data set contains 12794627 rows.
2. Our Data Cleaning steps include
   - Remove Categorical variables. As these variable does not help in characterization of DDoS/Flash/Normal traffic. In our case, we have removed IP, Timestamp, Protocol.
   - Remove those columns which have more than 50% data missing.
   - Remove all rows containing negative values as these are irrelevant.
   - Make a Correlation matrix of all the features.

- |Correlation| > 0.8 ———> Remove those features**.**

3. Randomly pick 10% of the data after applying the above steps.

4.2 Dimensionality Reduction

Dimensionality Reduction means reducing the input features in the training data set. The motive of the reduction matrix is to select the fewer features which are enough to classify the data and that generalize well and make the machine learning model simple and generalizable to other datasets. There are several ways to reduce the number of features. The Python library scikit-learn is the most widely used and provides fairly accurate feature importance. Ideally, each feature can be removed one by one, and then the permutation analysis can be performed to evaluate how many features are sufficient to retain the same accuracy, but that is very computationally expensive. Other techniques like Principal Component Analysis (PCA) can reduce the features but also makes the model opaque. It transforms the features into linear combinations which cannot be directly interpreted for describing the use case. Therefore, in this analysis, we intend to keep the individual features untransformed but pick the 10 most important ones.

As the scikit-learn feature importance works best on tree-based methods, this research work evaluated it on Random Forest as shown in Fig. 9.

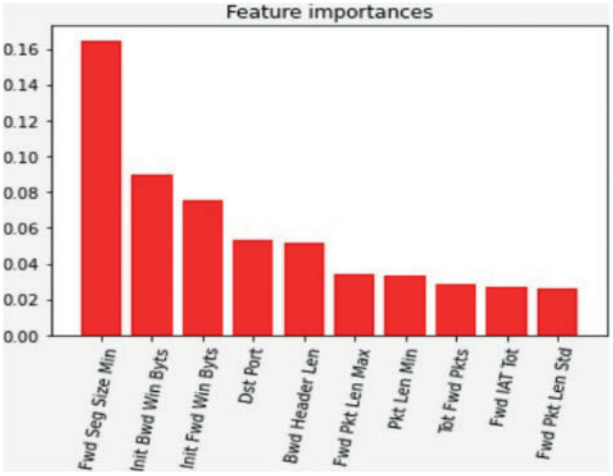

Figure 9: Top 10 feature list using scikit library

Machine Learning is usually referred to as "Black Box", as it remains hidden to a normal user about how a machine reached a particular decision [31]. What all features contributed to the decision-making. To better comprehend the features and their decisions, we calculate SHAP values for each data point. The Shapley value is the average estimated marginal contribution of one player. When each player may have contributed more or less than the others, Shapley value can help calculate a payout for all of them. Another library called Probatus was used to accomplish this. This library suggests leading features for discrimination of DDoS attack and normal traffic, and in addition to this, it also indicates feature's individual contribution towards DDoS attack and normal traffic identification. As shown in Fig. 10. Negative values on X-axis represent DDoS attack and positive values on x-axis contribute towards normal traffic. High value of Fwd Seg Size Min (Red Color) indicates it is a DDoS attack and low value of Fwd Seg Size Min (Blue color) suggests it is a normal traffic. Likewise, low value of

Init Fwd Win Bytes (Blue color) suggests it is a DDoS attack and High value of Init Fwd Win Bytes (Red Color) implies it is a normal traffic.

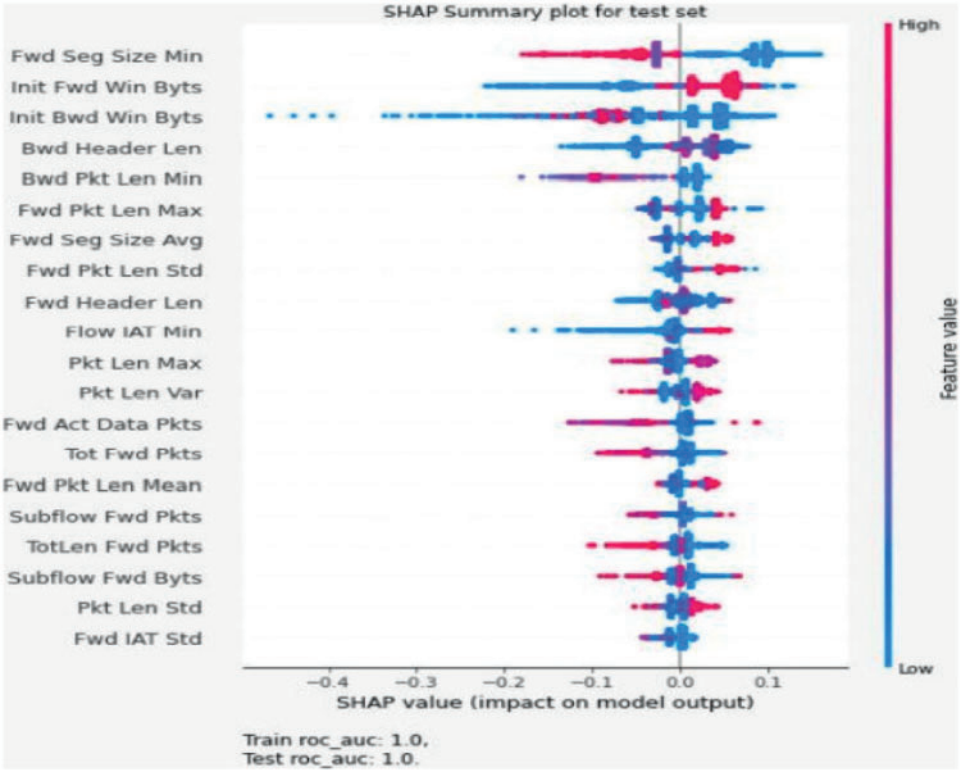

Figure 10: SHAP values for each data point in test set

The ten most prominent features with their description are shown in Tab. 3.

Table 3: Feature description

Feature name	Description
Fwd Seg Size Min	Minimum segment size observed in the forward direction
Init Bwd Win Byts	Total number of bytes sent in initial window in the backward direction
Init Fwd Win Byts	Total number of bytes sent in initial window in the forward direction
Dst port	Destination port address
Bwd Header Len	Total bytes used for header in the backward direction
Fwd Pkt Len Max	Maximum length of the packet in the forward direction
Pkt Len Min	Minimum length of the packet
Tot Fwd Pkts	Total number of packets in forward direction
Fwd Pkt Len Std	Standard deviation length of packet in forward direction

4.3 Performance Evaluation

Machine learning methods come in a variety of shapes and sizes. The main issue is determining which approach is optimal for our dataset [32]. The two major aims of an optimal DDoS security system are effectiveness and accuracy [33]. The confusion matrix, which is quantified in terms of the number of False Positives (FPs) and False Negatives (FNs), is used to evaluate the execution of each model (FNs). Predictive analysis for DDoS defense formulates a table called the confusion matrix as described in Tab. 4.

Table 4: Confusion matrix

Confusion matrix		Predicted condition	
		Normal	Attack
True condition	Normal	True negatives (TN)	False negatives (FN)
	Attack	False positives (FP)	True positives (TP)

Since we are interested in detecting DDoS, we call successful detection of DDoS in our data as "true positive" and the detection of normal as "true negative". Consequently, "false positive" would be when a data point is detected as DDoS but is normal. Similarly, a "false negative" would be when a data point is detected as normal but is a DDoS attack.

Precision, Recall, Accuracy, AUC, f1-score, Receiver Operating Characteristics (ROC) detection metrics to measure the performance of the proposed approach. Precision is a calculation of how much of the test data observed as attacks belongs to one of the attack groups. On the other hand, Recall is the ratio of detected attacks to the total attack events [34].

Recall =TP/ (TP + FN)

Precision= TP/ (TP + FP)

Accuracy= TP + TN/ (TP + FP +FN + TN)

Receiver Operator Characteristics (ROC): - This graph provides a simple way to summarize true positive and false positive rates.

AUC (Area Under Curve): - Allows for simple comparison of one ROC curve to another. The higher the AUC value, the better the model.

5 Results and Discussions

The results of our proposed algorithm with 10 features set in terms of ROC curve, False negatives, False positives, True Negatives, True Positives, Accuracy, Precision and Recall are shown in Fig. 11, Tabs. 5, 6.

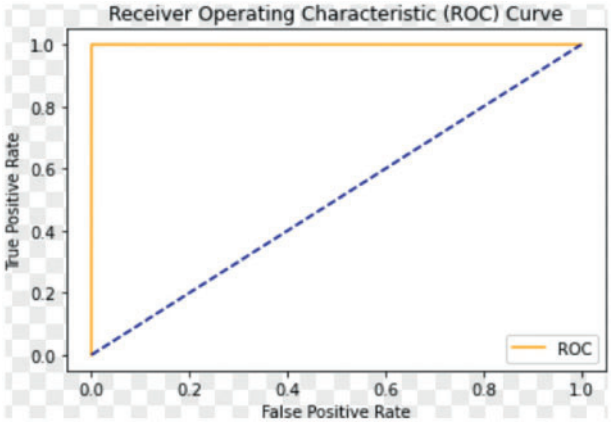

Figure 11: ROC curve for random forest

Table 5: TN, TP, FN, FP for random forest

Model	True negatives (TN)	True positives (TP)	False negatives (FN)	False positives (FP)
Random forest	251227	258526	49	56

Table 6: Results of random forest

Parameters	Precision		Recall		AUC	Accuracy
Model	DDoS	Normal	DDoS	Normal		
Random forest	0.99	0.99	0.99	0.99	0.999794214862008	0.999794060306

The results of the various machine learning models on 60 features data set are shown in Tab. 7. But challenge here is to process the voluminous data as a result, lot of computation is required by various machine learning models.

Table 7: TN, TP, FN, FP for machine learning models

S. no	Models	True negatives (TN)	True positives (TP)	False negatives (FN)	False positives (FP)
1	Logistic regression	246399	254855	4877	3727
2	Decision tree	251114	258420	162	162

(Continued)

Table 7: Continued

S. no	Models	True negatives (TN)	True positives (TP)	False negatives (FN)	False positives (FP)
3	K- Nearest neighbors	250564	258329	712	253
4	Random forest	251142	258493	134	89
5	Support Vector Machine (SVM)	249515	257499	1761	1083
6	NBC (Naive Bayes Classifier)	172911	245210	78365	13372

Fig. 12. Represents Receiver Operating Characteristics (ROC) for different machine learning models on the 60 features dataset.

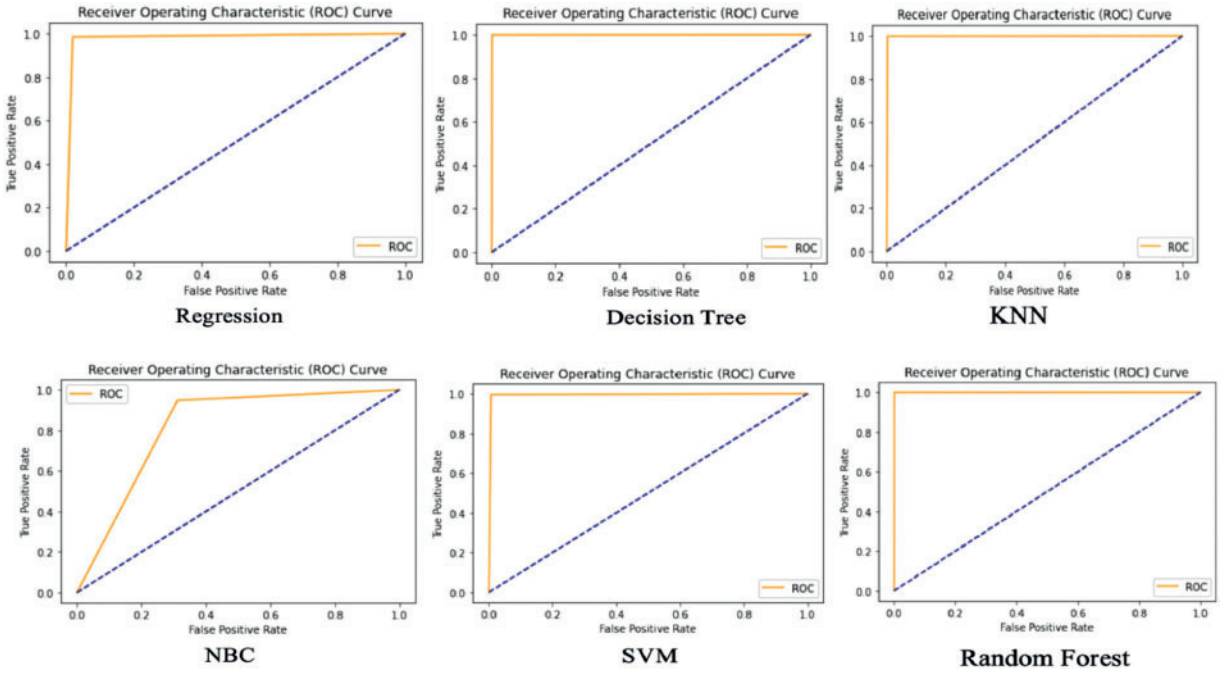

Figure 12: ROC curves for machine learning models

Tab. 8. Describes the comparative analysis of various machine learning models based on metrics like accuracy, recall, etc. It has been concluded that Random Forest performs best in all the machine learning models with 60 features data set.

Table 8: Comparative analysis of machine learning models based on metrics

Parameters	Precision		Recall		AUC	Accuracy
Models	DDoS	Normal	DDoS	Normal		
Logistic regression	0.9886	0.9888	0.9987	0.9876	0.98308892016427	0.9831247133
Decision tree	0.9884	0.9885	0.9888	0.9899	0.99936439843666	0.9993645289
K-Nearest neighbors	0.9885	0.9884	0.9888	0.9899	0.99809402466938	0.9981073161
Random forest	0.9886	0.9885	0.9888	0.9899	0.99956126851414	0.9995626331
Support Vector Machine (SVM)	0.9995	0.9995	0.9888	0.9899	0.99440177172359	0.9944219763
NBC (Naive Bayes Classifier)	0.76	0.93	0.76	0.93	0.81820948625814	0.8200734322

Above are the results of various machine learning models on 60 features data set. However, according to the "curse of dimensionality," the more characteristics in a data set, the greater the risk of the model overfitting the data. Because overfitted models can generalize well to out-of-time data, so, in this research work we have tried to minimize the input features without sacrificing the model's performance. This has been achieved by dimensionality reduction using 'feature importance' and SHAP value importance. Tab. 9. Represents comparative analysis of random forest model on 60 features data set with our proposed model.

Table 9: Comparison results

Model	True negatives	True positives	False negatives	False positives	Accuracy	Precision	Recall
Random forest	251142	258493	134	89	0.9995626331	0.988	0.988
Proposed algorithm	**251227**	**258526**	**49**	**56**	**0.999794060306**	**0.99**	**0.99**

It has been clear from the results that by reducing the features from 60 to 10, false positives and false negatives score decreases further as a result accuracy and recall improves.

6 Conclusion

Dimensionality Reduction applied to the existing data sets is an economical and effective method to improves accuracy and reduces the computational power needed for machine learning models.

Depending on the use case, practitioners may want to reduce false positives or false negatives in the model. Our model with reduced dataset shows that both false negatives and false positives are reduced as compared to the model trained on full dataset. Thus, avoiding the overfitting in model training by dimensionality reduction not only makes the model 'lite' which can be easily implemented on cloud systems, but its performance (both detection rate and precision) also improves.

In our future work, Firstly, we highly encourage to provide different datasets from different domains e.g., ecom, education, healthcare, etc. to be used to make this solution more generic. Secondly, we will try to use another feature dimensionality reduction on the same data set that not only reduces the feature set but also contributed towards decision making i.e., feature individual contribution towards DDoS attacks and Normal Traffic. Third, use of Auto ML, concept where machine trains and updates its model automatically, is encouraged for any future work, to take this concept to even one step further.

Acknowledgement: This work was supported by the Researchers Supporting Project (No. RSP-2021/395), King Saud University, Riyadh, Saudi Arabia.

Funding Statement: This work was supported by the Researchers Supporting Project (No. RSP-2021/395), King Saud University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Jinquan, M. A. Abdulhakim, A. Ali-Absi and H. J. Lee, "Analysis and protection of computer network security issues," in *22nd Int. Conf. on Advanced Communication Technology (ICACT)*, PyeongChang, Korea, 2020.

[2] K. S. Vanitha, S. V. Uma and S. K. Mahidhar, "DDoS: Attack techniques and mitigation," in *Int. Conf. on Circuits, Control and Communications(CCUBE)*, Bangalore, India, 2017.

[3] P. Nichalson, "Five most famous distributed denial of service attacks," *A10 Blog on Network Security*, vol. 63, pp. 13–18, 2021.

[4] F. S. Silva, E. Silva, E. P. Neto, M. Lemos, A. J. Neto *et al.,* "A taxonomy of DDoS attack mitigation approaches featured by SDN technologies in IoT scenarios," *Sensors*, vol. 20, no. 11, pp. 1–28, 2020.

[5] S. Chakraborty, P. Kumar and B. Sinha, "A study on DDoS attacks, danger and its prevention," *International Journal of Research and Analytical Reviews*, vol. 6, no. 2, pp. 10–15, 2019.

[6] T. Mahjabin, Y. Xiao and G. Sun, "A survey of a distributed denial-of-service attack prevention and mitigation techniques," *International Journal of Distributed Sensor Networks*, vol. 13, no. 12, pp. 1–33, 2017.

[7] R. Fouladi, O. Ermis and E. Anarim, "Anomaly based DDoS attack detection by using sparse coding and frequency domain," in *IEEE 30th Annual Int. Symp. on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Istanbul, India, 2019.

[8] P. Kaur, M. Kumar and A. Bhandari, "A review of detection approaches for distributed denial of service attacks,"*System Science and Control Engineering*, vol. 5, no. 1, pp. 301–320, 2017.

[9] R. S. Chaudhari and G. R. Talemale, "A review on detection approaches for distributed denial of service attacks,"in *Int. Conf. on Intelligent Sustainable Systems (ICISS)*, Palladam, India, 2019.

[10] A. Prakash, M. Satish, T. Bhargav and N. Bahlaji, "Detection and mitigation of denial-of-service attacks using stratified architecture," *Procedia Computer Science*, vol. 87, pp. 275–280, 2016.

[11] A. Bhandari, A. L. Sangal and K. K. Saluja, "Characterizing flash events and distributed denial-of-service attacks: An empirical investigation," *Security and Communication Networks*, vol. 9, no. 13, pp. 1–18, 2016.

[12] N. Hoque, D. K. Bhattacharyya and K. Kalita, "FFSC: A novel measure for a low rate and high-rate DDoS attack detection using multivariate data analysis," in *8th Int. Conf. on Communication Systems and Networks (COMSNETS)*, Banglore, India, 2016.
[13] M. A. Saleh and A. A. Manaf, "A novel protective framework for defeating HTTP based denial of service and distributed denial of service attacks," *Hindawi Publishing Corporation*, vol. 2015, pp. 1–19, 2015.
[14] X. Liu, X. Yang and Y. Lu, "To filter or to authorize network layer DoS defense against multimillion- node botnets," in *ACM SIGCOMM Computer Communications Review*, Seattle, USA, 2008.
[15] V. Deepa, K. M. Sudar and P. Deepalakshmi, "Detection of DDoS attack on SDN control plane using hybrid machine learning techniques,"in *Int. Conf. on Smart Systems and Inventive Technology (ICCSIT)*, Tirunelveli, India, pp. 299–303, 2018.
[16] A. Srivastava, B. B. Gupta, A. Tyagi, A. Sharma and A. Mishra, "A recent survey on DDoS attacks and defense mechanisms," *Communications in Computer and Information Science*, vol. 203, pp. 570–580, 2011.
[17] J. Markovic and P. Reiher, "A taxonomy of DDoS attacks and DDoS defense mechanisms," *ACM SIGCOMM Computer Communications Review*, vol. 34, no. 2, pp. 39–53, 2004.
[18] B. Benamar, K. Benamar, H. Fouzi and S. Ying, "DDoS attack detection using an efficient measurement-based statistical mechanism," *Engineering Science and Technology and International Journal*, vol. 23, no. 4, pp. 870–878, 2020.
[19] T. A. Hanger, "An effecting approach of detecting DDoS using neural networks,"in *Int. Conf. on Wireless Communications Signal Processing and Networking*, Chennai, India, 2017.
[20] S. N. Shiaeles, V. Katos, A. S. Karakos and B. Papadopoulos, "Real time DDoS detection using fuzzy estimators," *Computers and Security*, vol. 31, no. 6, pp. 782–790, 2012.
[21] A. S. Jose and L. R. Nair, "Mitigation of DDoS attacks over software defined network using machine learning and deep learning techniques," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 83, pp. 563–568, 2019.
[22] D. Prasad, B. Prasanta and C. Amarnath, "Machine learning DDoS detection using stochastic gradient boosting," *International Journal of Computer Sciences and Engineering*, vol. 7, no. 4, pp. 157–166, 2019.
[23] R. Roelofs and S. F. Keil, "A Meta-analysis of overfitting in machine learning," in *33rd Conf. on Neural Information Processing Systems*, Vancouver, Canada, 2019.
[24] F. Y. Osisanwo, J. E. T. Akinsola, O. Awodele, J. O. Hinmikaiye, O. Olakanmi *et al.,* "Supervised machine learning algorithms: Classification and comparison," *International Journal of Computer Trends and Technology*, vol. 48, pp. 128–138, 2017.
[25] J. Pérez-Díaz, I. A. Valdovinos, R. Choo and D. Zhu, "A flexible SDN based architecture for identifying and mitigating low-rate DDoS attacks using machine learning," in *IEEE Access*, vol. 4, pp. 155859–155872, 2016.
[26] B. V. Karan, D. G. Narayan and P. S. Hiremath, "Detection of DDoS attacks in software defined networks," in *3rd Int. Conf. on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, Bengaluru, India, pp. 265–270, 2018.
[27] S. Nanda, F. Zafari, C. DeCusatis, E. Wedaa and B. Yang, "Predicting network attack patterns in SDN using machine learning approach," in *IEEE Conf. on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Palo Alto, USA, pp. 167–172, 2016.
[28] F. Silveira, A. Medeiros, B. Junior, G. Solar and F. Silveira, "Smart detection: An online approach for DoS/DDoS attack detection using machine learning," *Security and Communication Networks*, vol. 2019, pp. 1–15, 2019.
[29] C. Li, Y. Wu, Z. Sun, W. Wang and X. Li, "Detection and defense of DDoS attack based on deep learning in openflow - based SDN," *International Journal of Communication Systems*, vol. 31, no. 5, pp. 1–15, 2018.
[30] M. S. Elsayed, N. Khac, S. Dev and A. D. Jurcut, "Machine learning techniques for detecting attacks in SDN," in *IEEE 7th Int. Conf. on Computer Science and Network Technology*, Dalian, China, 2019.
[31] H. Patel, D. S. Rajput, G. T. Reddy, C. Iwendi, A. K. Bashir *et al.,* "A review on classification of imbalanced data for wireless sensor network," *International Journal of Distributed Sensor Networks*, vol. 16, no. 4, pp. 1–15, 2020.

[32] M. Shafiq, Z. Tian, A. K. Bashir, X. Du, and M. Guizani, "Corrauc: A malicious bot-iot traffic detection method in iot network using machine learning techniques," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3242–3254, 2021.

[33] C. Iwendi, M. Uddin, J. Ansere, P. Nkurunziza, J. H. Anajemba *et al.,* "On detection of sybil attack in large scale VANETs using spider- monkey technique," *IEEE Access*, vol. 6, pp. 47258–47267, 2018.

[34] S. Rehman, M. Khaliq, S. I. Imtiaz, A. Rasool, M. Shafiq *et al.,* "DIDDOS: An approach for detection and identification of DDoS cyberattacks using gated recurrent units," *Future Generation Computer Systems*, Elsevier, vol. 118, pp. 453–466, 2021.