JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

DDoS (Distributed Denial of Service) Attack Detection Using Machine Learning

Mr.S.R.Naresh [1], N. Naveen Kumar[2], R. Mukesh Kumar[3], T. Praveen Raj[4]P.Gnana sekar[5]

[1]Associate Professor, Department of Electronics and Communication Engineering,

[2345]Final Year Students, Department of Electronics and Communication Engineering,

K.L.N. College of Engineering.

as losses in finance and Data. Due to the unexpected nature of DDoS attacks, conventional methods of detection often prove inadequate. Machine learning (ML) algorithms have emerged as a promising solution for detecting DDoS attacks by recognizing patterns and irregularities in network traffic that may indicate an attack. This article offers a comprehensive overview of DDoS attack detection utilizing ML, covering a range of algorithms, including supervised, unsupervised, and deep learning. The article also discusses crucial features required to train ML models for detecting DDoS attacks, such as packet size, packet rate, and network flow features. Furthermore, this paper evaluates various machine learning algorithms such as random forest, cat-boost classifier, and gradient boosting by predicting their accuracy.

***IndexTerms* - DDoS attack, Random Forest, Gradient Boosting, Cat-Boosting**

I. INTRODUCTION

A Distributed Denial of Service (DDoS) attack is an unauthorized attempt to disrupt the regular operation of a website. These attacks can be disastrous for businesses and organizations because they can cause significant downtime, financial loss, and harm to their reputation. Detection is essential for preventing harmful DDoS attacks. DDoS attacks can be detected using various methods and tools, ranging from simple traffic analysis to more complex machine learning algorithms. Analyzing traffic patterns and detecting anomalies makes it simple to detect DDoS attacks and take the necessary precautions to mitigate their effects. DDoS attacks can be recognized, and proper action should be taken to lessen their effects. In this situation, it is crucial to have a thorough awareness of the different forms of DDoS attacks as well as the available detection methods. With the use of this knowledge, organizations can proactively monitor their systems, spot potential risks, and take quick action to keep their networks and systems secure. DDoS attack detection and mitigation is a difficult undertaking, but machine learning can be a helpful tool for locating these attacks and taking the necessary steps to lessen their impact.

Distributed Denial of Service (DDoS) :

Attacks known as distributed denial of service (DDoS) are frequent cyberthreat that can seriously harm businesses and their online services. In a DDoS attack, a large group of hacked systems or devices, known as a botnet, are used to overload a targeted network, server, or website with a tremendous quantity of traffic and prevent normal users from accessing it. By examining network traffic data, machine learning techniques like Random Forest, Gradient Boosting, and Cat-Boost have been used to identify DDoS attacks. These algorithms can spot trends and irregularities in the traffic data that point to a DDoS assault, including an unusual spike in traffic from a particular IP address or an increase in the number of requests.

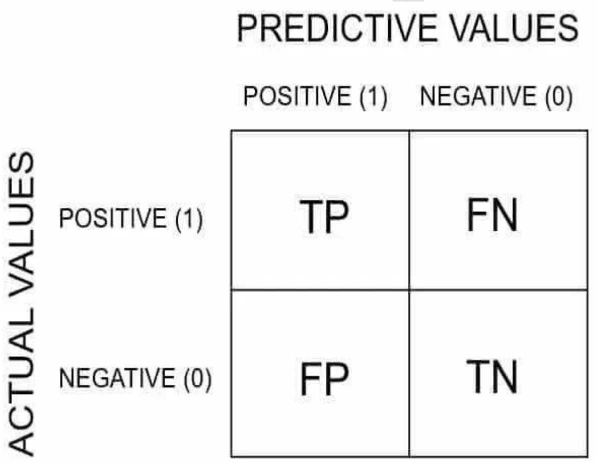
CORRELATION MATRIX:

A correlation matrix in machine learning is a table that displays the pairwise correlations between several variables in a dataset. The correlation coefficient between any two variables is represented by each cell in the matrix. The degree and direction of the linear link between two variables are measured by the correlation coefficient. For a several reasons, a correlation matrix can be helpful in machine learning. For instance, it can be used to determine which variables have a strong correlation with one another, which may indicate duplicate or unimportant features. The effectiveness of machine learning models can be enhanced by removing highly correlated characteristics, which can aid in reducing overfitting. Statistical software programs like Python's NumPy and Pandas libraries can be used to compute correlation matrices.

CONFUSION MATRIX :

N is the total number of target classes, and a confusion matrix is a N x N matrix used to assess how well a classification model is working. The machine learning model's predictions are put up against the actual target values in a matrix. Our classification model's overall performance and the kind of mistakes it is making are now clear to us.

- **True Positive (TP):** When the expected value or the expected class matches the observed value. The model anticipated a positive result, and the actual value was positive.
- **True Negative (TN):** Either the expected value or the predicted class matches the actual value. The model anticipated a negative result, but the actual value was negative. The expected value was incorrectly predicted.
- **False Positive (FP) :**The model projected a positive number, but the actual value was negative. The type I error is another name for it. The expected value was incorrectly predicted.
- **False Negative (FN):** The model projected a negative result, while the actual value was positive. Type II mistake is another name for it.



		PREDICTIVE VALUES	
		POSITIVE (1)	NEGATIVE (0)
ACTUAL VALUES	POSITIVE (1)	TP	FN
	NEGATIVE (0)	FP	TN

II. XISTING SYSTEM

MACHINE LEARNING TECHNIQUES USED:

i. SUPPORT VECTOR MACHINES (SVMS) :

Support Vector Machines (SVMs) are a group of machine learning algorithms with supervised learning that are used for regression and classification. SVMs work by locating the ideal "hyperplane," or border, that divides the data points into distinct classes. An SVM algorithm looks for a hyperplane that maximizes the margin between the two classes in a binary classification task. The margin is the separation between each class's nearest data points and the hyperplane. The biggest margin hyperplane is selected by the SVM algorithm because it is less likely to incorrectly categorize new data points. A kind of SVM created expressly for classification issues is Support Vector Classification (SVC). SVC works by transforming the input data into a higher-dimensional spacewhere the data can be separated by a hyperplane.

ii. The Restricted Boltzmann Machine:

Unsupervised machine learning algorithms for feature extraction and dimensionality reduction include the Restricted Boltzmann Machine (RBM). RBMs are a particular kind of artificial neural network that can autonomously learn a probability distribution over the input data. RBM is a potent unsupervised machine learning technique that has the ability to learn intricate representations of input data and extract crucial characteristics. It is helpful for many applications, including DDoS attack detection, because of its capacity forperforming dimensionality reduction.

iii. The Geometric Mean Decomposition Support Vector Machine:

A machine learning algorithm called the Geometric Mean Decomposition Support Vector Machine (GDSVM) combines the Geometric Mean Decomposition (GMD) and the Support Vector Machine (SVM) algorithms. For classification tasks, GDSVM is especially beneficial for high-dimensional data sets with unbalanced classes. For classification tasks, the GDSVM method, a potent machine learning technique, includes the GMD and SVM algorithms. It is a helpful tool for many applications, including the detection of DDoS attacks, due to its capacityto handle wide data with classified.

III. SYSTEM MODEL

Acquiring and preprocessing network traffic data, extracting pertinent features, training machine learning, assessing performance, and deploying the models in a real-time environment are all steps in a system model for detecting DDoS attacks.

1. Data collection

This involves gathering information about network traffic from multiple devices, including switches, routers, and firewalls. The data may consist of details like flow characteristics, IP addresses, and packet headers.

2. Pre-processing

The collected data is pre-processed to extract relevant features and remove noise. This can include techniques such as filtering, normalization, and feature selection.

3. Feature extraction

This involves extracting meaningful features from the pre-processed data. This can be done using techniques such as statistical analysis, machine learning algorithms, and pattern recognition.

4. Model training

The extracted features are used to train machine learning models for DDoS attack detection. This can include algorithms such as Random Forest, Cat-Boost, and Gradient Boosting.

5. Model evaluation

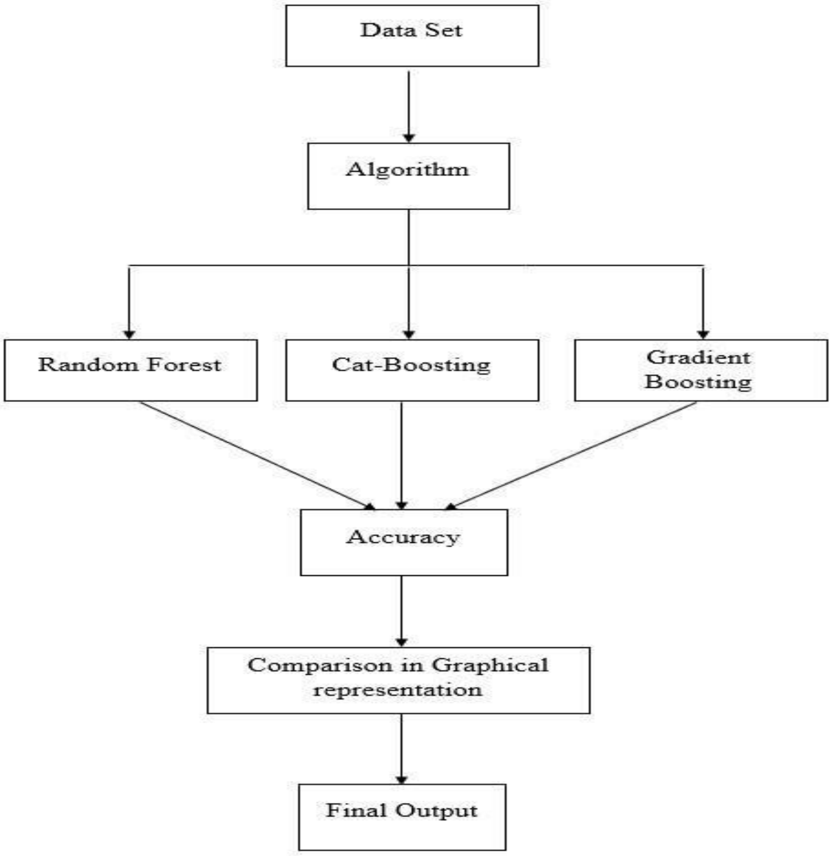
The trained models are evaluated using performance metrics as accuracy. This step is important to ensure that the models are effective in detecting DDoS attacks and have low false positive rates.

6. Deployment

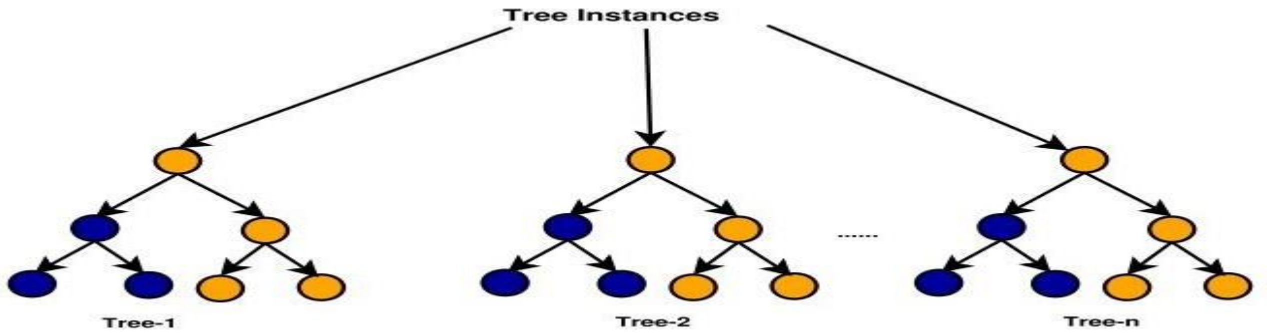
The trained model are deployed in a real-time environment for continuous monitoring of network traffic. This can include deploying the models on edge devices, such as routers and firewalls, or in a cloud-based environment.

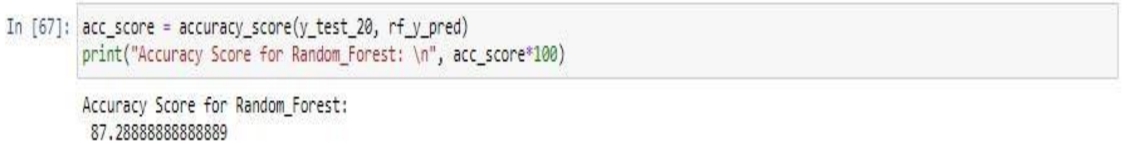
Flow Diagram

Random Forest:



A well-liked machine learning method called Random Forest is utilized for both classification and regression problems. Several decision trees are combined in this ensemble learning technique to produce predictions. The robust and adaptable Random Forest method has the accuracy and generalization ability to handle huge and complicated datasets. The technique builds a forest of decision trees, each of which is trained using a randomly chosen portion of the training data and features. Each decision tree in the set is trained using a random subset of the input data, and the algorithm uses a random subset of the available features at each point. This randomization enhances the model's accuracy by lowering overfitting. The Random Forest algorithm aggregates all of the forest's trees' predictions during the prediction phase to get the final forecast. For classification issues, the algorithm determines the mode of all the anticipated classes, and for regression problems, the mean or median of the expected values. When compared to other machine learning methods, Random Forest provides a number of benefits. It can handle high-dimensional data with numerous features and missing values, and feature scaling is not necessary. Additionally, it can offer feature importance scores that can be used to comprehend the relative weights given to various features during the prediction process. Overall, Random Forest is a powerful and flexible machine learning algorithm that can be applied to a wide range of tasks and is suitable for both beginners and experts in machine-learning.



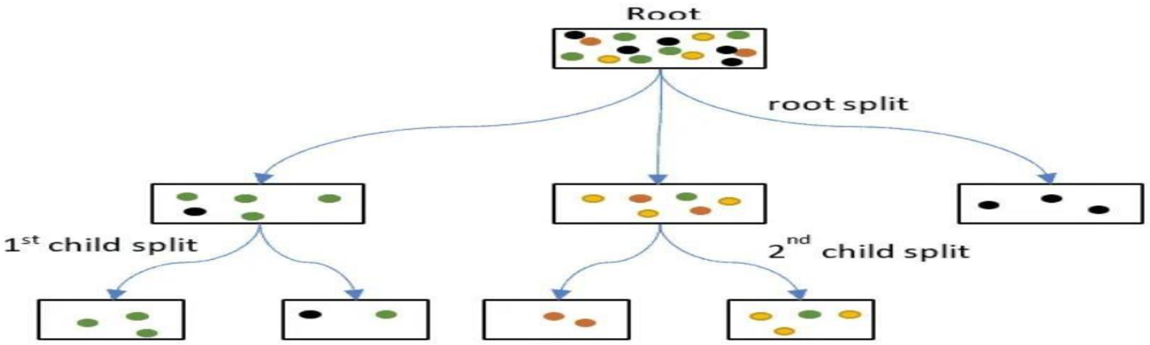


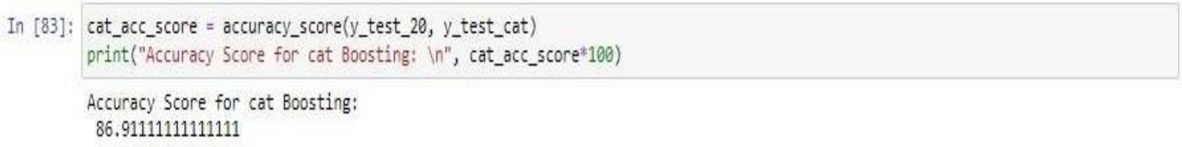
```
In [67]: acc_score = accuracy_score(y_test_20, rf_y_pred)
         print("Accuracy Score for Random_Forest: \n", acc_score*100)

         Accuracy Score for Random_Forest:
          87.28888888888889
```

Fig: Random Forest Accuracy

Cat-Boosting:

The gradient boosting method of framework, on which Cat-Boost is based, involves iteratively adding weak learners to a model in order to increase its predictive potential. However, Cat-Boost employs a cutting-edge strategy known as "ordered boosting" in contrast to conventional gradient boosting algorithms to handle category variables more skillfully. Using an ordering mechanism, ordered boosting transforms categorical variables into numerical values while maintaining the relationships between the categories. Application areas where Cat-Boost has been utilized successfully include fraud detection, predicting customer turnover, and picture categorization. Cat-Boost can be used to categorize network traffic data and precisely identify attacks in the context of DDoS attack detection. It is a helpful tool for analysis because it can handle category variables and handle missing data making it a useful tool for analyzing network traffic data, which often contains a mix of numerical and categorical variables.



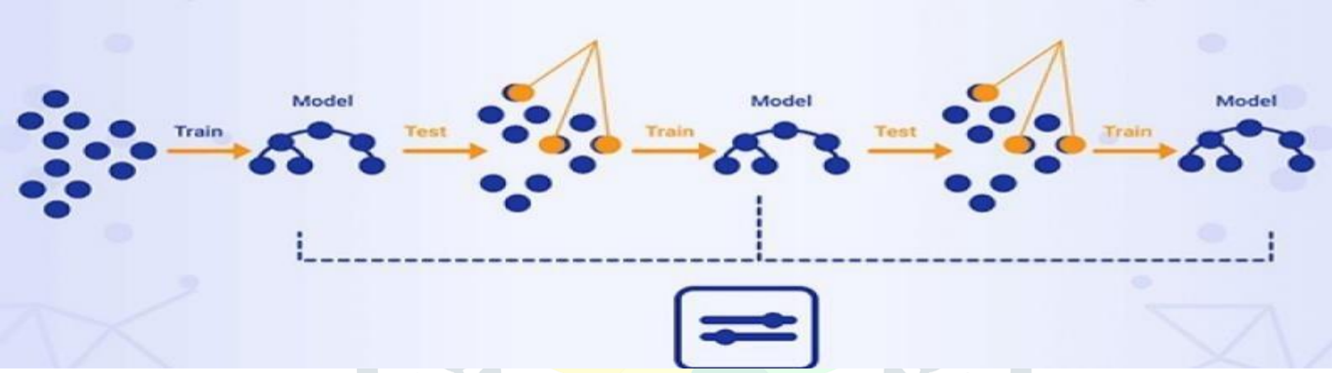


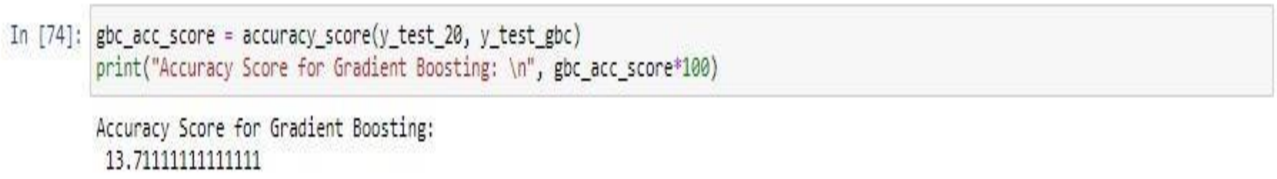
```
In [83]: cat_acc_score = accuracy_score(y_test_20, y_test_cat)
         print("Accuracy Score for cat Boosting: \n", cat_acc_score*100)

         Accuracy Score for cat Boosting:
          86.91111111111111
```

Fig: Cat Boosting Accuracy

Gradient Boosting:

Building a series of decision trees in which each new tree is trained to anticipate the residual error of the prior tree is how the method operates. The approach iteratively enhances the model's performance in training by using gradient descent to optimize a loss function. In order to minimize the loss function, it, in other words, modifies the weights of the decision trees based on the difference between the expected output and the actual output. Gradient Boosting's strength rests in its capacity to merge weak prediction models into powerful and precise models. Each decision tree in the series gains knowledge from the residual mistakes of the previous tree, which lowers the model's bias and variance and produces better accuracy. There are several advantages to using Gradient Boosting. It can handle missing values, outliers, and skewed data, and can also handle categorical and numerical data. Additionally, it can provide feature importance scores, which can help in understanding the importance of each feature in the prediction process. Gradient Boosting can be computationally costly and prone to overfitting, though, if the number of trees or their depth is excessively high. To obtain the best performance, hyperparameter adjustment is crucial. Overall, Gradient Boosting is a robust and popular machine-learning technique that may be utilized for a wide range of tasks. Many data scientists and analysts find it to be an invaluable tool because of its capacity to manage complicated and heterogeneous data andgenerate precise predictions.

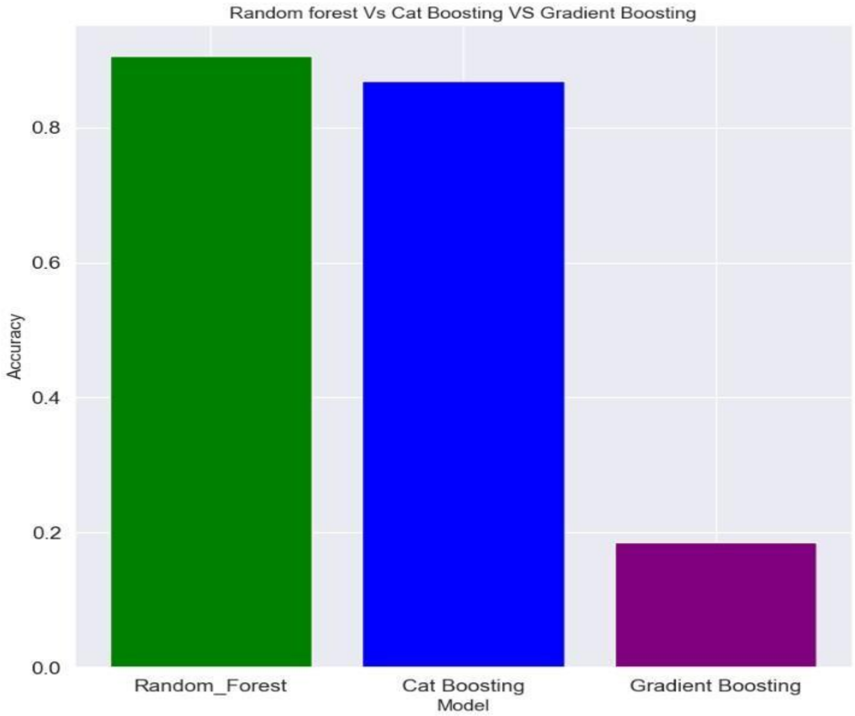




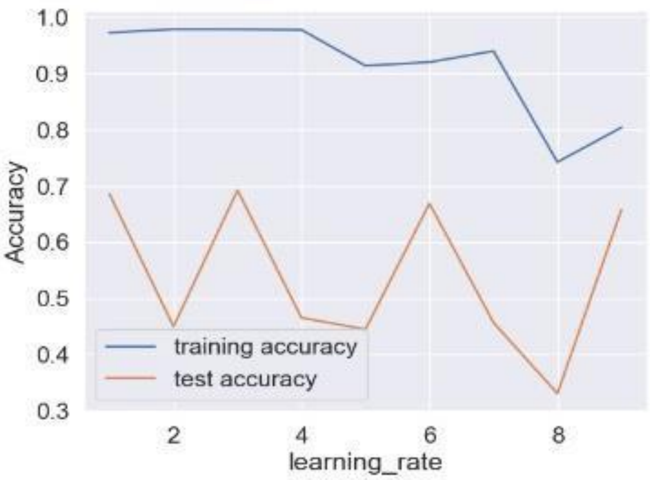
```
In [74]: gbc_acc_score = accuracy_score(y_test_20, y_test_gbc)
         print("Accuracy Score for Gradient Boosting: \n", gbc_acc_score*100)

         Accuracy Score for Gradient Boosting:
          13.7111111111111
```

Fig: Gradient Boosting Accuracy

IV. RESULT



We have trained the Random Forest algorithm which has high accuracy than comparing other algorithms like Cat Boosting and Gradient Boosting. So we conclude that our trained Random Forest model is the best model for DDoS detection. Cat Boosting can also be used for DDoS detection since it has an accuracy of 86.91



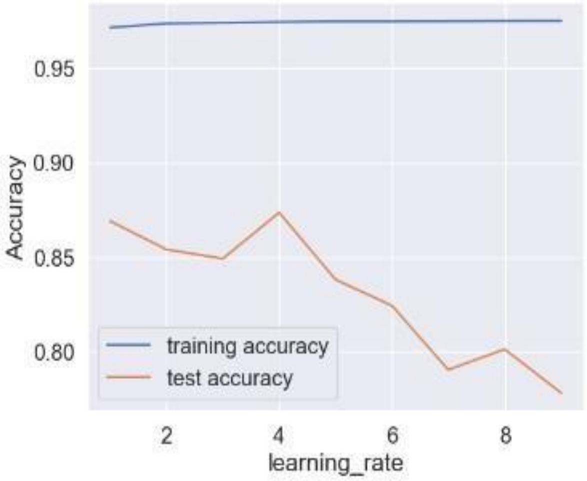
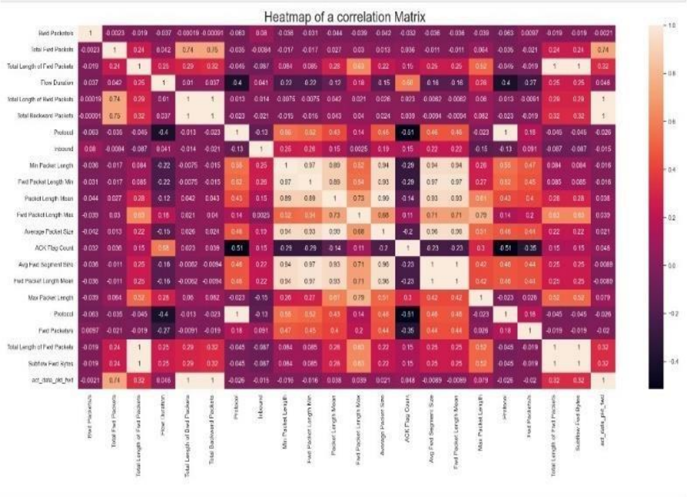


Fig: Gradient Boosting **Fig: Cat-Boosting**

Table : Algorithm & Accuracy

Algorithm	**Accuracy**
Random Forest	**87.288888888889**
Cat Boosting	**86.911111111111**
Gradient Boosting	**13.711111111111**



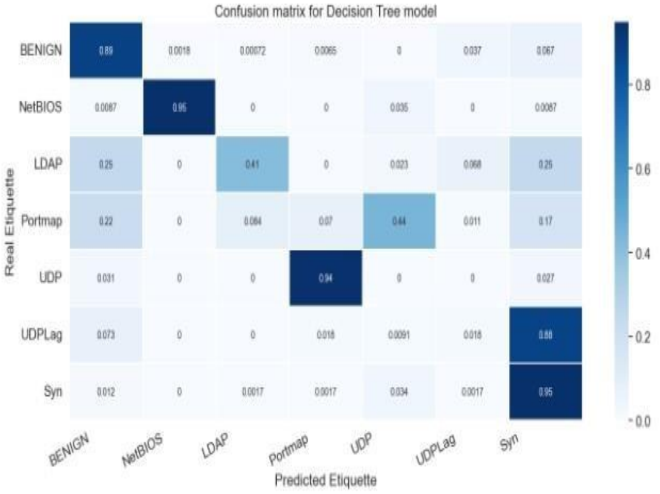


Fig: Heatmap of Correlation matrix

Fig: Confusion matrix for Decision Tree

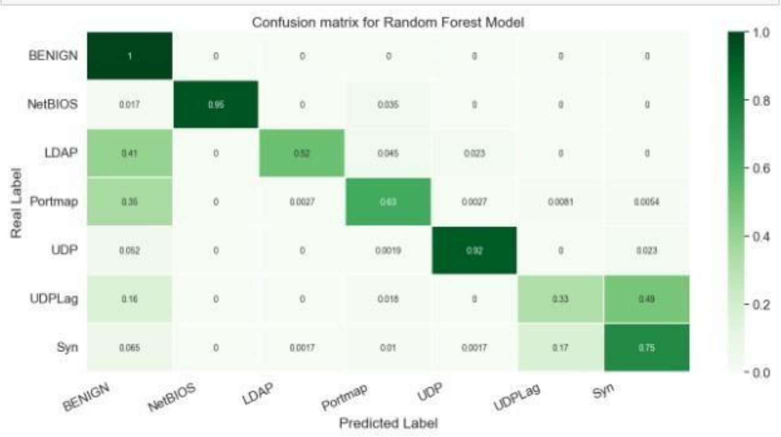


Fig: Confusion matrix for Random Forest

V. CONCLUSION

DDoS attacks pose a substantial threat to modern networks, and prompt detection is essential to avoiding severe service and infrastructure interruptions. The ability of machine learning algorithms to quickly evaluate vast amounts of network traffic data and precisely identify patterns and abnormalities related to attacks makes them a promising tool for DDoS attack detection. The ability of machine learning algorithms to rapidly evaluate huge amounts of network traffic data and precisely identify trends and attacks makes them a promising tool for DDoS attack detection. Deep learning developments will probably make DDoS attack detection more sophisticated and precise.

Future Work :

- The future implementation will focuses on implementing greater accuracy
- Implement a real-time system analysis to prevent denial of service

VI. REFERENCES

1. S. Garg, K. Kaur, N. Kumar and J. J. P. C. Rodrigues, "Hybrid Deep-Learning-Based Anomaly Detection Scheme for Suspicious Flow Detection in SDN: A Social Multimedia Perspective," in IEEE Transactions on Multimedia, vol. 21, no. 3, pp. 566-578, March 2019, doi: 10.1109/TMM.2019.2893549.

2. M. Zekri, S. E. Kafhali, N. Aboutabit and Y. Saadi, "DDoS attack detection using machine learning techniques in cloud computing environments," 2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech), Rabat, Morocco, 2017, pp. 1-7, doi: 10.1109/CloudTech.2017.8284731.

3. "A Novel Method for Detecting DDoS Attacks Based on Random Forest Algorithm" by Meng Liu, Shengli Liu, andHuaqi Wang, in IEEE Access, 2019.

4. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," in IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41-50, Feb. 2018, doi: 10.1109/TETCI.2017.2772792.

5. Hussain, Q. Du, B. Sun and Z. Han, "Deep Learning-Based DDoS-Attack Detection for Cyber–Physical System Over 5G Network," in IEEE Transactions on Industrial Informatics, vol. 17, no. 2, pp. 860-870, Feb. 2021, doi: 10.1109/TII.2020.2974520.

6. " Iqbal, H., Ali, S., & Aslam, F. (2021). DDoS Attack Detection using CatBoost Algorithm with Feature Selection and Dimensionality Reduction. International Journal of Advanced Computer Science and Applications (IJACSA), 12(2), 18-23. doi: 10.14569/IJACSA.2021.0120203