International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

DDoS Attack Classification Using Machine Learning

Keerthana N
Department of Computer Engineering, Sri Jayachamarajendra College of Engineering, JSS University, Mysore, Karnataka

Vismaya C
Department of Computer Engineering, Sri Jayachamarajendra College of Engineering, JSS University, Mysore,

Dr.S MANJULA
Assistant Professor. Department of Computer Science and Engineering, JSS Science and Technology University.

Abstract – DDoS (Distributed Denial of Service) attacks pose a significant threat to network security by overwhelming target systems with excessive traffic, rendering them inaccessible to legitimate users. Traditional rule-based detection methods struggle to keep up with evolving attack patterns, making machine learning (ML) a promising alternative. This study explores various ML models, including Random Forest, K- Nearest Neighbors (KNN), XGBoost, and Decision Tree classifiers, to detect and classify DDoS attacks effectively. Using a dataset containing over 852,585 network traffic records, we preprocess the data through feature selection, encoding, and outlier removal before training and evaluating different models. Performance is assessed using metrics such as accuracy, F1-score, recall, and confusion matrices. The results indicate that the Random Forest classifier achieves the highest accuracy (99.82%), followed closely by Decision Tree and KNN. The study highlights the effectiveness of ML in DDoS detection and suggests future improvements, such as hyperparameter tuning, deep learning techniques, and real-time deployment for enhanced security.

Index Terms – Random Forest, KNN, SVM, Decision Tree.

1. INTRODUCTION

The rapid advancement of digital technologies has significantly increased global connectivity, making cyberattacks more frequent and sophisticated. Among the most severe threats is the **Distributed Denial-of-Service (DDoS) attack**, which disrupts online services by overwhelming a target system with excessive malicious traffic. These attacks can cripple businesses, financial institutions, and even critical infrastructure, leading to substantial economic and operational losses. Traditional rule-based intrusion detection systems (IDS) have limitations in identifying modern DDoS attacks, as attackers continuously evolve their tactics to bypass static security measures. Moreover, rule-based systems require constant updates and manual configuration, making them inefficient in handling large-scale and dynamic attacks.

Machine learning (ML) provides a promising alternative for **automated and adaptive DDoS detection**. By analyzing network traffic patterns and distinguishing between normal and malicious activities, ML-based models can identify attack signatures in real time, reducing the time taken to detect and mitigate threats. In this research, we implement multiple ML algorithms, including **Random Forest, Decision Tree, Logistic Regression, Support Vector Machine, and K- Nearest Neighbors (KNN)**, to classify network traffic and detect DDoS attacks with high accuracy.

This study aims to evaluate the effectiveness of different machine learning models by analyzing their performance in terms of **accuracy, recall, F1-score, and confusion matrices**. The findings will contribute to the development of more robust, scalable, and real-time DDoS detection systems, enhancing cybersecurity defenses in modern networks.

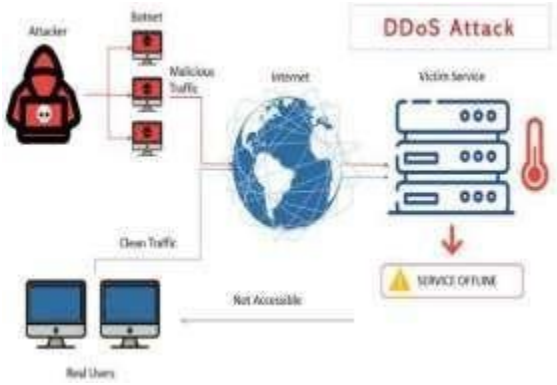


Fig 1: DDoS attack, where an attacker, using a botnet, overwhelms a victim service, rendering it inaccessible to real users

2. BACKGROUND AND RELATED WORK

2.1. Literature Survey

Kimmi Kumari & M. Mrunalini (2022) proposed a mathematical model for detecting DDoS attacks using Logistic Regression and Naive Bayes classifiers. They employed the CAIDA 2007 dataset and implemented their model using the Weka data mining platform. Their study highlights the effectiveness of ML-based detection compared to traditional rule-based systems. Their findings demonstrated that Logistic Regression achieved higher accuracy for detecting low-volume attacks, while Naive Bayes was more effective for high-volume attacks. The study also pointed out the limitations of dataset imbalance and suggested improvements through feature selection techniques. Additionally, their research emphasized the need for real-time detection capabilities and recommended hybrid models integrating both statistical and deep learning approaches for better detection accuracy and reduced false positives.

Wang et al. (2021) introduced an ensemble learning method combining Decision Trees and Support Vector Machines (SVM) for anomaly detection in network traffic. Their study demonstrated that combining multiple classifiers improves overall detection accuracy and reduces false positives. By leveraging an ensemble approach, the model effectively distinguished between benign and malicious traffic patterns, addressing the challenges posed by evolving DDoS tactics. The research further emphasized the benefits of feature extraction techniques, which played a critical role in improving the model's predictive power. Wang et al. also analyzed the impact of network traffic variations on model performance and found that incorporating adaptive learning mechanisms could further enhance detection in dynamic network environments.

Behal & Kumar (2017) introduced an entropy-based DDoS detection mechanism, leveraging statistical randomness to distinguish between legitimate and attack traffic. Their method continuously monitors entropy values, raising alerts when deviations indicate an ongoing attack. While lightweight and real-time, it struggles to differentiate legitimate traffic bursts from actual threats and may not generalize well due to predefined thresholds. In contrast, Mehdi et al. (2011) proposed a flow-based detection approach, utilizing SDN controllers to monitor traffic surges and detect unusual spikes in flow statistics. Although effective in identifying high-volume attacks, this method relies on manually defined thresholds, making it prone to false positives or negatives, and imposes a high processing load on the controller. Comparatively, entropy- based detection is efficient but lacks robustness in complex traffic s cenarios, while flow-based detection isSDN -compatible but resource-intensive.

Modern solutions integrate machine learning, adaptive thresholding, and hybrid techniques to improve detection accuracy, mitigate false alarms, and reduce overhead, addressing the limitations of these earlier methods.

Jia et al. (2022) proposed FORT, a lightweight DDoS detection scheme for SDN environments that enhances detection efficiency while reducing computational overhead. Traditional SDN-based DDoS detection methods rely on periodically requesting flow rules from switches, which is both CPU- intensive and can congest communication channels. To address this, FORT distributes the detection process to edge switches and activates detection based on periodic port state retrieval rather than constant flow rule requests. It employs ARIMA (Auto Regressive Integrated Moving Average) for adaptive port monitoring and SVM (Support Vector Machine) for attack detection. Experimental results show that FORT significantly reduces controller load, achieving a false alarm rate of just 0.039% compared to 1.24% in traditional schemes, while also reducing southbound channel load by over 60%. Additionally, the model effectively detects multiple attack types and even previously unseen attack patterns, demonstrating its robustness. Future work aims to further refine traffic classification by distinguishing malicious and legitimate traffic to improve overall network security.

Songa & Karri (2024) proposed an integrated SDN framework for early detection of DDoS attacks in cloud environments, addressing the limitations of traditional controller-level detection. Their RDAER model shifts detection to SDN switches, leveraging traffic clustering, anomaly prediction, and event correlation to identify attack patterns more efficiently. The framework integrates Recursive Feature Elimination (RFE) for feature selection, Density-Based Spatial Clustering (DBSCAN) for grouping similar traffic, and time-series techniques like ARIMA, Lyapunov exponent, and exponential smoothing for anomaly detection. A rule-based classifier at the controller level correlates detected anomalies and classifies traffic as either normal or malicious. When a switch detects a DDoS attack, the countermeasure module updates the flow table to mitigate the attack, ensuring a proactive defense mechanism. Evaluated on the CICD DoS 2019 dataset, RDAER achieved 99.92% accuracy and a fast detection time of 20 seconds, outperforming existing methods. Future improvements aim to further refine training and testing data to enhance model accuracy and adaptability in diverse SDN-cloud environments.

Yu et al. (2021) introduced a cooperative DDoS attack detection scheme for SDN that combines entropy-based monitoring and ensemble learning to enhance detection accuracy while reducing controller overhead. Their approach distributes detection tasks between edge switches (data plane) and the controller (control plane). The edge switch hosts a coarse-grained preliminary detection module, which uses entropy to monitor network traffic in real time and report anomalies to the controller. If an anomaly is detected, the fine- grained detection module at the controller level further examines traffic using an ensemble learning model, specifically Random Forest, to classify the traffic accurately. This method efficiently offloads computational tasks from the controller by utilizing the idle computing power of edge switches, reducing southbound communication overhead and CPU utilization at the control plane. Simulation results on ICMP and SYN flood attacks demonstrate that this approach enables fast and accurate DDoS detection while maintaining scalability. As network size increases, the scheme effectively reduces controller workload and shortens peak CPU utilization, making it a practical and scalable solution for DDoS mitigation in SDN environments.

Clinton et al. (2024) proposed a deep learning-based DDoS attack classification method for SDN environments, leveraging image transformation and convolutional neural networks (CNNs) to enhance detection accuracy. Given the growing threat of IoT botnet-driven DDoS attacks, especially against the centralized SDN controller, early detection is crucial to prevent network disruptions. Their method converts captured network traffic into image data, which is then classified using a CNN- based model. The research evaluates performance on a custom test-bed dataset and two benchmark datasets (CTU-13 and In SDN), achieving over 99% classification accuracy. The model's effectiveness was compared against VGG19, InceptionV3, Efficient Net, ResNet, and DenseNet, outperforming them in accuracy, precision, recall, F1-score, and MCC. This approach demonstrates high reliability in distinguishing DDoS traffic from normal traffic and is a promising technique for automated SDN security. Future work includes real-time traffic analysis, integrating other detection models, and developing a prevention system to block attack sources and mitigate DDoS threats in SDN environments

Nanda et al. (2020) proposed an ML-based framework using deep learning, enhancing detection accuracy. Tang et al. (2019) employed decision trees and random forests for attack classification, though ML methods demand high computational resources.

.

Sekar et al. (2008) and Zhang (2013) developed lightweight detection approaches that minimize controller load. Jia et al. (2022) introduced FORT, a lightweight DDoS detection scheme that spreads rule-based detection across edge switches. It utilizes ARIMA for adaptive port monitoring and SVM for attack detection, significantly reducing false alarms (0.039%) and southbound channel load (60%). FORT improves detection efficiency while maintaining network stability.

Salmi & Oughdir (2023) conducted a performance evaluation of deep learning techniques for detecting DoS attacks in wireless sensor networks (WSNs). They highlighted the increasing use of WSNs for data collection and monitoring in various industries, emphasizing their vulnerability to security threats due to limited computational resources. Traditional intrusion detection systems (IDS) struggle to counter evolving and complex DoS attacks effectively.

To address these challenges, the authors trained multiple deep learning-based IDS models on the WSN-DS dataset, which contains data on four major types of DoS attacks: Blackhole, Gray hole, Flooding, and Scheduling attacks. Their study assessed how well deep learning models could identify these threats in resource-constrained WSN environments. The findings demonstrated that deep learning techniques significantly improved detection accuracy and adaptability compared to conventional methods.

Additionally, the research highlighted the need for further improvements, including optimizing IDS models to reduce false positives and enhance real-time attack detection. The study suggested refining feature selection techniques and leveraging hybrid models combining deep learning with other AI-based methods for better security and efficiency in WSNs.

3. Existing Methodology

Several methodologies have been proposed over the years for detecting DDoS attacks, ranging from statistical models to advanced machine learning and deep learning approaches. Early methods used classifiers like Logistic Regression and Naive Bayes, which showed effectiveness for low- and high-volume attacks respectively, while ensemble techniques combining Decision Trees and SVMs improved detection accuracy and reduced false positives. Entropy-based and flow- based models provided lightweight real-time detection but struggled with adaptability and resource constraints.

Recent innovations include SDN-based models like FORT and RDAER, which distribute detection to edge devices and use time-series analysis and clustering for improved accuracy and lower overhead. Deep learning approaches, such as CNN-based image transformation and IDS models for wireless sensor networks, have demonstrated high performance in complex environments. Overall, modern methods focus on hybrid, scalable, and adaptive systems to address the evolving nature of DDoS attacks effectively.

4. Proposed Methodology for DDoS Attack Detection

The proposed methodology consists of multiple stages, including data preprocessing, feature selection, model training, and evaluation. The framework is designed to efficiently detect and classify DDoS attacks using machine learning techniques.

Data Preprocessing:

- Cleaning the dataset by removing null values and outliers.
- Encoding categorical features and performing normalization.
- Visual analysis using graphs to understand the distribution of "Malicious" vs. "Benign" traffic and detect patterns in protocol types, IP addresses, and request durations.

Feature Selection:

- Applying correlation matrices and referencing previous studies (e.g., using Neighbourhood Component Analysis) to identify the most relevant features.
- This step reduces dimensionality and enhances model efficiency.

Model Training:

The models used include:

- Random Forest – Provides high accuracy and handles high-dimensional data effectively.
- Decision Tree – Simple and interpretable; used as a baseline.
- K-Nearest Neighbours (KNN) – Effective but computationally intensive.
- Logistic Regression – Suitable for binary classification.
- Support Vector Machine (SVM) – Effective for high-dimensional spaces and clear decision boundaries.

Model Evaluation:

- Performance assessed using accuracy, precision, recall, F1-score, and confusion matrices.
- Feature selection improved performance across models, with Random Forest achieving the highest accuracy (up to 99.99%).

Comparative Analysis:

- The methodology also reviewed various existing research efforts, comparing traditional statistical techniques, ensemble methods, deep learning, and SDN-based approaches.
- Highlights include entropy-based detection, flow-based analysis, CNN-based image transformation, and hybrid systems combining edge and controller-level detection

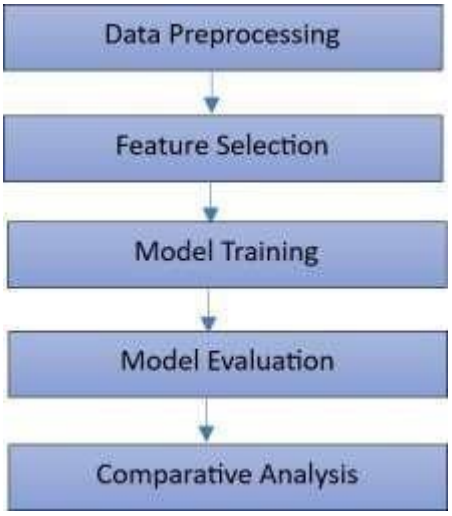


Fig 2: Flowchart of the proposed methodologies

4. Result and Analysis

4.1 Libraries

- Pandas: Purpose of using pandas is for data manipulation and analysis. It Functions for reading/writing CSV, Excel, SQL, and JSON file and also supports operations like filtering, grouping, merging, and pivoting data.
- NumPy: It is used for Numerical computing and array manipulation, provides multi-dimensional arrays, Supports mathematical operations like linear algebra, Fourier transforms, and random number generation.
- Seaborn: It's a Statistical data visualization library, built on top of Matplotlib for enhanced visuals, also Supports complex visualizations like heatmaps, violin plots, and pair plots.
- Scikit-Learn (sklearn): To implement machine learning and predictive modelling methods, provides tools for data preprocessing, model selection, and evaluation, Supports classification, regression, clustering, and dimensionality reduction.

4.2 Data Visualization and Data Analysis

Data visualization is the process of representing data graphically to make it easier to understand, analyses, and communicate insights. It helps in identifying patterns, trends, and outliers in data.

Data Analysis is the process of inspecting, cleaning, transforming, and modelling data to extract useful information, draw conclusions, and support decision-making. It is widely used in various industries like business, healthcare, finance, and technology.

The detailed analysis of the data is shown below in the form of graphs and plots.

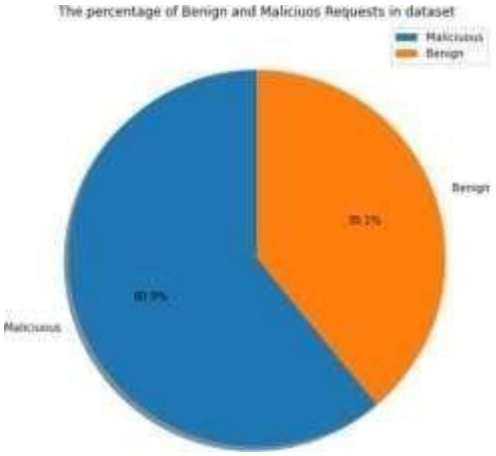


Fig 3: Visualization of "Malicious" and "Benign" requests in a dataset using a pie chart.

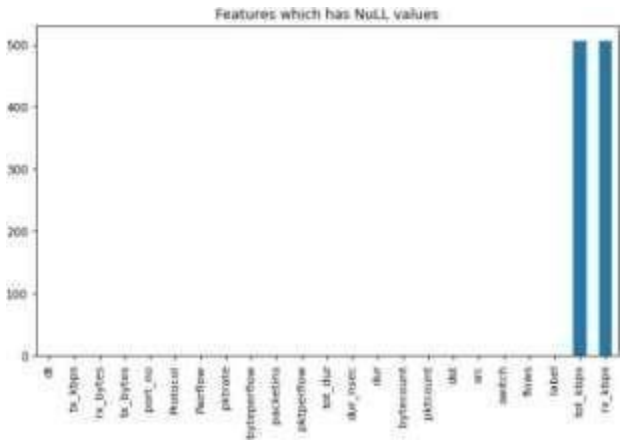


Fig 4 Null Valued Features

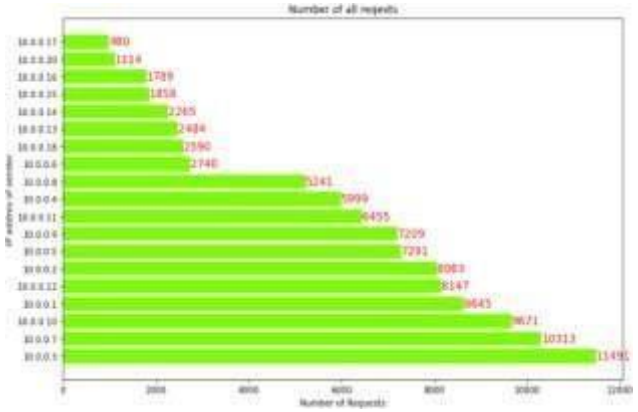


Fig 5: Plotting the graph to identify the number of all requests and IP address of sender.

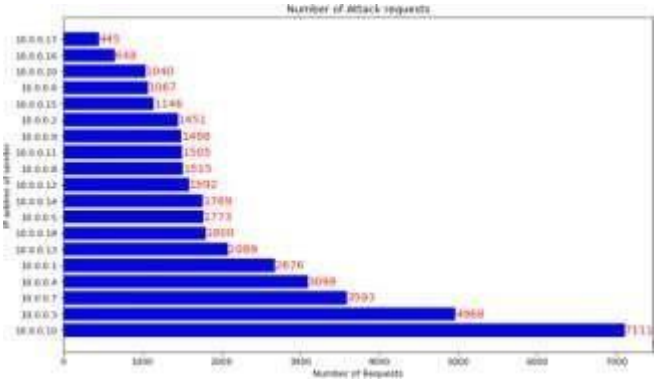


Fig 6: representing number of attack requests from different IP addresses

s.

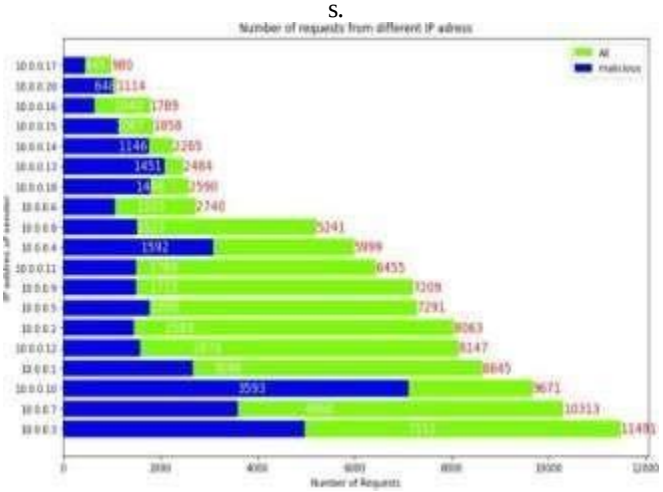


Fig 7: Comparative graph of number of attacks and requests from the users.

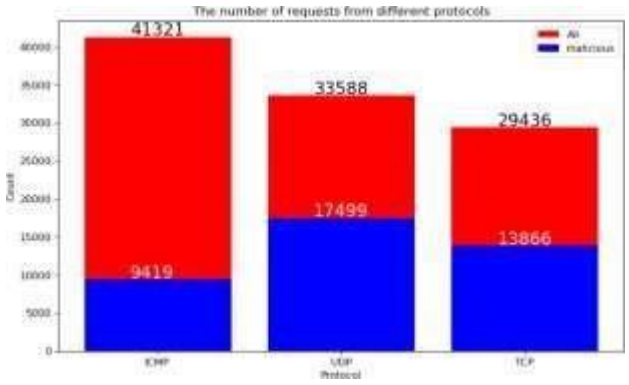


Fig 8: The number of requests from different protocols.

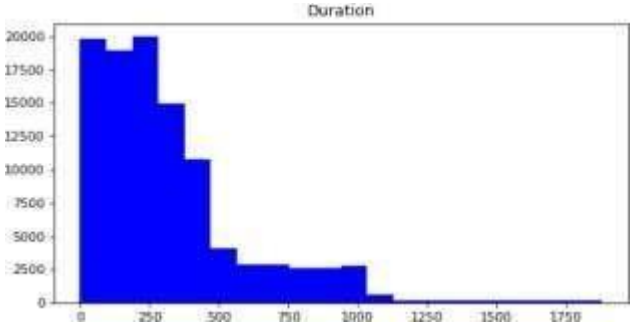


Fig 9: Duration of the requests.

4.3 Model Training & Selection

Multiple machine learning algorithms are trained and compared:

- **Random Forest Classifier**: Random Forest is an ensemble learning method that constructs multiple decision trees during training and combines their predictions to enhance accuracy. It mitigates overfitting by averaging the outputs of individual trees, thus improving generalization. This model is highly effective in handling large datasets and high-dimensional feature spaces, making it suitable for detecting network anomalies. Random Forest also provides feature importance rankings, which can be useful in identifying the most influential network parameters for DDoS detection.

- **K-Nearest Neighbors (KNN)**: KNN is a non-parametric, distance-based classifier that classifies network traffic based on the similarity to existing data points. It calculates the distance between a new data point and its k-nearest neighbors in the feature space and assigns a label based on the majority class among those neighbors. KNN is particularly effective in capturing attack patterns when the dataset is well- structured and labelled. However, it has a high computational cost, especially when dealing with large-scale network traffic, as it requires storing the entire dataset for classification.

- **Logistic Regression:** Logistic Regression is a supervised machine learning algorithm used for classification problems, particularly binary classification (i.e., problems where there are only two possible outcomes). Unlike linear regression, which predicts continuous values, logistic regression predicts the probability that an input belongs to a particular class.

- **Support Vector Machine:** Support Vector Machine (SVM) is a supervised learning algorithm used for classification and regression tasks. It is widely used for binary classification problems and can handle high- dimensional data effectively. The main objective of SVM is to find the optimal decision boundary that best separates data points of different classes.

- **Decision Tree Classifier**: A Decision Tree is a rule-based classifier that splits data into hierarchical structures based on feature values. It is simple to interpret and implement, making it a valuable baseline model for comparison with more advanced techniques. Decision Trees can efficiently classify network traffic but tend to overfit when trained on large datasets. This issue can be addressed by pruning techniques, which remove unnecessary branches to improve generalization. Despite its simplicity, Decision Trees provide valuable insights into the feature importance of DDoS attack detection.

4.4 Model Evaluation and results

After training and making predictions with Logistic Regression, it's essential to evaluate its performance using various **metrics** like precision, F1-score, recall.

Explanation of metrics:

4.4.1 Precision: Precision is a key metric used to evaluate the performance of a classification model.

Precision is defined as:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

Where:

- True Positives (TP) – Correctly predicted positive cases
- False Positives (FP) – Incorrectly predicted positive cases

4.4.2 F1-score (f1_score): Harmonic mean of precision and recall.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.4.3 Recall (recall_score): Measures how many actual positive instances were correctly classified:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

4.5 Model Evaluation Results**:**

Accuracy of various ML Algorithms without using feature selection.

a. Logistic Regression:

```
Accuracy: 76.64%

####################################################################
Fitting :Best solver is :  liblinear
criterio####################################################################
The Accur            precision    recall  f1-score   support
#########

           0       0.84      0.79      0.81     20824
           1       0.66      0.72      0.69     11128

    accuracy                           0.77     31152
   macro avg       0.75      0.76      0.75     31152
weighted avg       0.77      0.77      0.77     31152

accur
macro
weighted

########################################################################
--- 56.05--- 9.088924272537231 seconds --- time for LogisticRegression
```

b. Support Vector Machine:

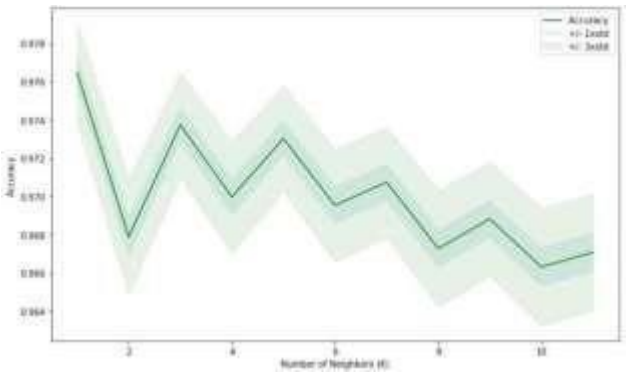
```
Accuracy: 78.48%
####################################################################
Accuracy: 96.53%
####################################################################
Accuracy: 96.67%
####################################################################
Accuracy: 54.52%
####################################################################
Accuracy of SVM model 97.0%

####################################################################
best kernel is :  rbf
####################################################################
              precision    recall  f1-score   support

           0       0.97      0.98      0.97     18750
           1       0.97      0.95      0.96     12402

    accuracy                           0.97     31152
   macro avg       0.97      0.96      0.97     31152
weighted avg       0.97      0.97      0.97     31152

####################################################################
--- 1334.302262544632 seconds ---
```

c. Decision Tree:

```
Accuracy of RF is : 99.99%

####################################################################
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     18984
           1       1.00      1.00      1.00     12168

    accuracy                           1.00     31152
   macro avg       1.00      1.00      1.00     31152
weighted avg       1.00      1.00      1.00     31152

####################################################################
--- 23.751417636871338 seconds ---
```

d. Random Forest:

e. KNN:



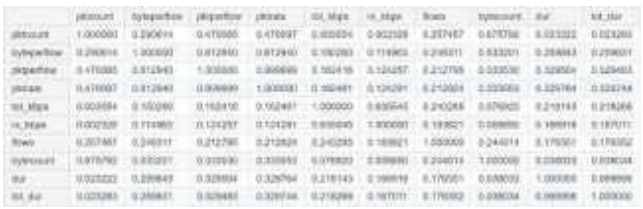
Prediction of various ML algorithms by using feature selection:

The feature selection is done by using weights allocated by referring to the results analyzed in an article titled Machine Learning Approach Equipped with Neighborhood Component Analysis for DDoS Attack Detection in Software-Defined Networking.

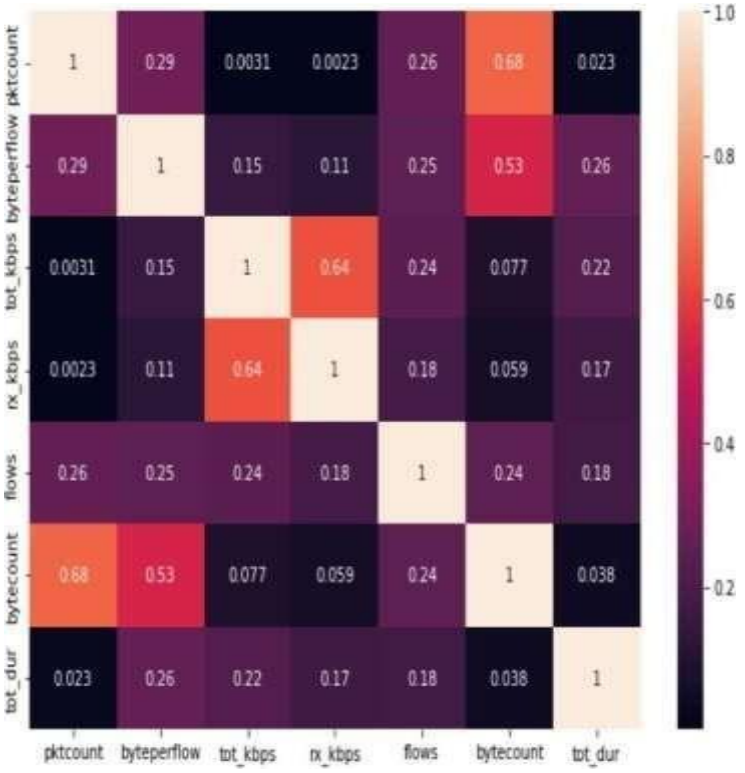
a. The selected features:

	features	weights
0	src	17.87
1	pktcount	15.16
2	dst	13.64
3	byteperflow	12.97
4	pktperflow	11.35
5	pktrate	11.35
6	tot_kbps	9.68
7	rx_kbps	9.66
8	flows	8.95
9	bytecount	4.92
10	dt	2.33
11	Protocol	1.31
12	dur	1.11
13	tot_dur	1.11

The correlation matrix is used to compute the numerical relationship in the dataset is shown below



b. Heatmap showing the correlation of the dataset after feature selection.

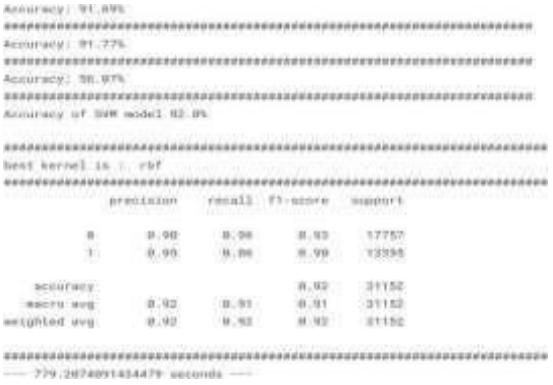


c. The accuracy of various algorithms:

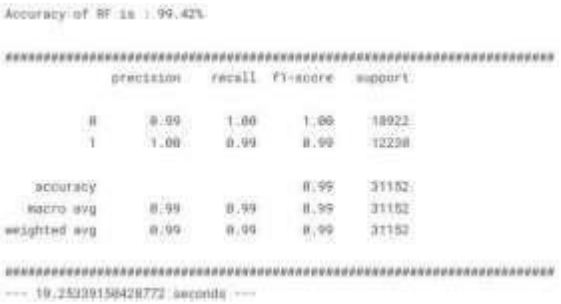
- Logistic Regression:

```
Accuracy: 75.21%

######################################################################
Best solver is :  sag
######################################################################
              precision    recall  f1-score   support

           0       0.85      0.77      0.81     20967
           1       0.60      0.72      0.65     10185

    accuracy                           0.75     31152
   macro avg       0.72      0.74      0.73     31152
weighted avg       0.77      0.75      0.76     31152


######################################################################
--- 2.9728949069976807 seconds --- time for LogisticRegression
```

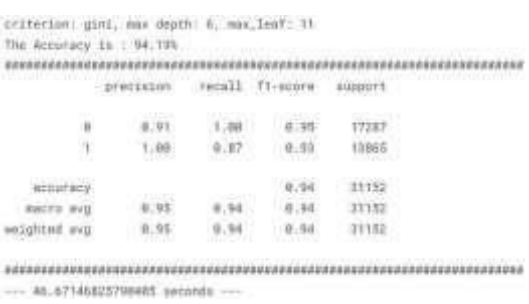
- SVM model:



- Random Forest:



- Decision Tree:



5. CONCLUSION

This study demonstrates the effectiveness of machine learning models in detecting and classifying DDoS attacks with high accuracy. By analyzing network traffic data and leveraging multiple ML algorithms—Logistic Regression, Random Forest, Decision Tree, K-Nearest Neighbors (KNN), and SVM —we achieved an impressive detection accuracy of up to 99.99%. Among the models tested, Random Forest and Decision Tree performed the best, highlighting their robustness in identifying attack patterns.

The results confirm that machine learning-based approaches significantly outperform traditional rule-based intrusion detection systems, which struggle with evolving attack patterns. The data preprocessing techniques, including feature selection, and correlation, played a crucial role in improving model performance. Additionally, the study emphasizes the importance of using multiple performance metrics, such as precision, F1-score and recall to comprehensively evaluate model effectiveness.

In conclusion, machine learning proves to be a powerful tool for automated and scalable DDoS detection, offering improved security for modern networks. Implementing such models in real-time Intrusion Detection Systems (IDS) can strengthen cybersecurity defenses, minimizing the risks posed by cyberattacks.

6. FUTURE WORK:

- Hyperparameter tuning and deep learning techniques to further enhance accuracy.
- Real-time implementation with streaming data for faster attack mitigation.
- Adaptive learning mechanisms to handle evolving attack patterns dynamically.

7. REFERENCES:

- Kun Jia, Chaoge Liu, Qixu Liu, Junnan Wang, Jiazhi Liu & Feng Liu - A lightweight DDoS detection scheme under SDN context **https://cybersecurity.springeropen.com/articles/10.1186/s42400-022-00128-7**
- Asha Varma Songa & Ganesh Reddy Karri - An integrated SDN framework for early detection of DDoS attacks in cloud computing. https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-024-00625-9
- Shanshan Yu, Jicheng Zhang, Ju Liu, Xiaoqing Zhang, Yafeng Li & Tianfeng Xu - A cooperative DDoS attack detection scheme based on entropy and ensemble learning in SDN

https://jwcn-eurasipjournals.springeropen.com/articles/10.1186/s13638-021-01957-9

- Urikhimbam Boby Clinton, Nazrul Hoque & Khumukcham Robindro Singh - Classification of DDoS attack traffic on SDN network environment using deep learning https://cybersecurity.springeropen.com/articles/10.1186/s42400-024-00219-7
- Rishikesh Sahay a b,Gregory Blanc a b, Zonghua Zhang b c, Hervé Debar a b - ArOMA: An SDN based autonomic DDoS mitigation framework.

https://www.sciencedirect.com/science/article/abs/pii/S0167404817301499

- Balasubramanian V, Aloqaily M, Reisslein M (2021) An SDN architecture for time sensitive industrial IoT. Comput Netw 186:107739
- Ramprasath J, Seethalakshmi V (2021) Improved network monitoring using software-defined networking for ddos detection and mitigation evaluation. Wireless Pers Commun 116(3):2743–275

Article Google Scholar

- Gadallah WG, Omar NM, Ibrahim HM (2021) Machine learning-based distributed denial of service attacks detection technique using new features in software- defined networks. Int J Comput Netw Inform Secur 13(3):15–27

Google Scholar

- Rawat SG, Obaidat MS, Pundir S, Wazid M, Das AK, Singh DP, Hsiao KF (2023) A Survey of DDoS Attacks Detection Schemes in SDN Environment. In 2023 International Conference on Computer, Information and Telecommunication Systems (CITS) (pp. 01–06). IEEE
- Dong S, Abbas K, Jain R (2019) A survey on distributed denial of service (DDoS) attacks in SDN and cloud computing environments. IEEE Access 7:80813– 80828

Article Google Scholar

- Dahiya A, Gupta BB (2020) Multi attribute auction based incentivized solution against ddos attacks. Comput Secur 92:101763

Article Google Scholar

- Samaan SS, Jeiad HA (2023) Feature-based real-time distributed denial of service detection in SDN using machine learning and Spark. Bullet Electric Eng Inform 12(4):2302–2312

Article Google Scholar