DDos attack detection method based on machine learning

Yeefong Wu

School of letter and science, University of Wisconsin-Madison, Madison, 53703, US

Wu564@wisc.edu

Abstract. This paper introduces a novel machine learning-based approach for the detection of Distributed Denial of Service (DDoS) attacks. The proposed method employs three different classifiers, namely Support Vector Machine (SVM), Random Forest, and Multilayer Perceptron (MLP), to accurately classify network traffic as normal or malicious. The approach incorporates various features extracted from network traffic, such as packet length, packet inter-arrival time, and destination port number, to train the classifiers. The results demonstrate high accuracy rates, with Random Forest outperforming the other classifiers in terms of detection accuracy. This proposed method offers a promising solution for detecting DDoS attacks in real-time and has the potential to be integrated into existing network security systems. The issue of DDoS attacks is becoming increasingly critical, with the proliferation of connected devices and the growing dependence on the Internet. There is an urgent need for advanced techniques to detect and mitigate such attacks. Machine learning approaches have shown great potential in identifying anomalous behavior and detecting DDoS attacks in real-time. The proposed method is a promising step towards achieving this goal. Furthermore, the proposed approach has practical applications in the context of network security. By integrating this method into existing security systems, it will enhance the system's ability to detect and prevent DDoS attacks. The approach can be implemented in different network environments, making it versatile and applicable to a variety of settings. In conclusion, the proposed machine learning-based approach provides a robust and effective solution for the detection of DDoS attacks. It is capable of accurately classifying network traffic as normal or malicious in real-time, and has the potential to enhance the overall security of networking systems.

Keyword: DDoS attack, detection, machine learning, SVM, random forest, MLP.

1. Introduction

In today's increasingly connected world, internet security is a critical concern for businesses and individuals who rely heavily on digital networks for communication, information sharing, and commerce. With the growth in the use of internet resources, the number and complexity of security threats have also increased, including Distributed Denial-of-Service (DDoS) attacks, fraudulent information propagation, and the challenges of applying laws and regulations to internet-based activities. This article explores these threats and the use of machine learning algorithms, such as SVM, random forest, and decision trees, to combat them [1].

DDoS attacks are among the most common types of attacks used by hackers to compromise the security of internet resources. Attackers flood victim networks with traffic, consuming network bandwidth and system resources. These attacks can be launched from many hosts, making them challenging to trace and stop. One of the difficulties in mitigating DDoS attacks is distinguishing legitimate traffic from malicious traffic. Machine learning algorithms can help by analyzing traffic patterns and classifying traffic based on flow features and information entropy, identifying anomalies in traffic and differentiating between legitimate users and attackers [2]. Fraudulent information propagation is another significant threat to internet security. Attackers can use email or other forms of communication to propagate fraudulent information that violates laws and regulations set by governing bodies. Bayesian principles and machine learning algorithms, such as decision trees, Boosting, K-nearest neighbors, and Support Vector Machines, are effective in combating this threat. These algorithms analyze data based on multivariate, polynomial, and Boolean attribute types and identify patterns in data to classify it. Traffic classification is an area where machine learning algorithms have shown particular promise in combating internet security threats. By analyzing flow features and information entropy, machine learning algorithms can identify anomalies in network traffic and distinguish between legitimate users and attackers. For example, support vector machines (SVMs) have been effective in classifying network traffic based on flow features such as packet size, duration, and direction, while random forests have been used to classify network traffic based on features such as packet payload and transport protocol [3]. Anomaly detection is another area where machine learning algorithms can be effective in improving network security. Invisible detection includes identifying differences in patterns of network behavior, which can indicate potential security threats such as attempted intrusions or malware. One method for detecting anomalies is to use machine learning algorithms to classify network connections as normal or abnormal based on flow characteristics, packet content, or other properties. . For example, decision trees are used to identify network anomalies based on characteristics such as packet size, protocol, and source/destination IP address [4].

Machine learning algorithms can also be used to identify malicious network traffic based on its informational content. For example, machine learning algorithms can be used to analyze network traffic content, such as email messages, to identify fraudulent information or other harmful content. This can be very useful against phishing attacks, which are a common type of internet security threat. Machine learning algorithms such as Bayesian classifiers and decision trees can be used to analyze email messages and identify patterns that indicate phishing attempts.

In addition to improving network security, machine learning algorithms can also be used to improve network management efficiency and effectiveness. For example, machine learning algorithms can be used to predict network traffic patterns and adjust network policies and configurations. This can help optimize network performance and reduce the risk of network congestion and downtime. Machine learning algorithms can also be used to automate routine network management tasks, such as network monitoring and device configuration, freeing network administrators to focus on more strategic tasks [5].

By analyzing traffic and data, machine learning algorithms can also be used in network management and network architecture to identify anomalies and classify traffic based on flow feature and information entropy. This can help network administrators detect and respond to security threats quickly and effectively. One technology that has been particularly effective in enhancing network security is Software-Defined Networking (SDN). SDN enables greater control and flexibility in network architecture, allowing administrators to quickly adjust network policies and configurations in response to changing security threats.

Despite the potential benefits of machine learning algorithms in combating internet security threats, there are also challenges to their use. One challenge is the need for accurate data to train algorithms. Another challenge is the need for real-time analysis, which can require significant computational resources. Finally, machine learning algorithms are only as effective as the features they use to classify data. This means that identifying the right features and feature weights is critical to the success of machine learning algorithms in combating internet security threats [6].

In conclusion, internet security is a critical concern in today's increasingly connected world. Distributed Denial-of-Service (DDoS) attacks, fraudulent information propagation, and the challenges of applying laws and regulations to internet-based activities are some of the most significant threats to internet security. Machine learning algorithms such as SVM, random forest, and decision trees are effective tools for analyzing data, classifying traffic, and detecting anomalies. However, the challenges of data quality, computational resources, and feature identification must be carefully considered in order to effectively combat internet security threats. Technologies such as SDN have also shown promise in enhancing network security, but it is important to ensure that they are properly configured and maintained to ensure their effectiveness. So machine learning algorithms are a powerful tool for enhancing internet security and improving network management. By analyzing data, classifying traffic, and identifying anomalies, machine learning algorithms can help detect and respond to security threats quickly and effectively [7]. However, the effectiveness of machine learning algorithms depends on the quality of training data, the accuracy of feature identification, and the availability of computational resources. As the threat landscape continues to evolve, it will be critical to continue to develop and refine machine learning algorithms to ensure that they remain effective in combating internet security threats [8].

2. Related technologies

2.1. Support vector machine

Support Vector Machines (SVM) have been widely used in the field of network security for detecting Distributed Denial of Service (DDOS) attacks. SVM is a powerful machine learning algorithm that is well-suited for identifying patterns in high-dimensional, noisy, and complex data, making it a strong choice for detecting network anomalies [9].

One reason why SVM is a good choice for DDOS attack detection is its ability to handle high-dimensional network traffic data effectively. In a DDOS attack, network traffic data can be extremely complex and high-dimensional, and SVM can learn to distinguish between normal and anomalous traffic patterns even in such high-dimensional spaces. Additionally, SVM is robust to noisy data and can handle imbalanced data, which is important when dealing with network traffic data that can be noisy and may contain outliers. SVM can also effectively model non-linear relationships between input variables, which is important when dealing with complex network traffic patterns that may not be linearly separable. In addition to these technical advantages, SVM has been shown to achieve high accuracy in many classification tasks, including DDOS attack detection. High accuracy can help reduce false positives and false negatives, which are critical in identifying and mitigating a DDOS attack [10].

The overall work flow is listed below, As shown in Figure 1:

Preliminary data: The traffic data in the network must be cleaned, processed and converted into the appropriate format. This will include handling missing values, evaluating features, and dividing the data into training and testing.

Feature selection: Select a group of relevant features that are most useful for detecting DDOS attacks. This can help reduce problem size and improve SVM performance.

Model training: Train the SVM model using the selected features. The SVM algorithm aims to find a hyperplane that optimally separates normal and anomalous traffic patterns. In practice, this involves solving optimization problems to find hyperplane nonuniformities that minimize misclassification.

Model evaluation: Test the performance of the SVM model on a separate test. This can help determine whether the model fits the training data and provide an estimate of its performance.

Model tuning: Fine-tune the hyperparameters of the SVM model to improve its performance. This may include kernel updates, regular updates, or other benchmarks.

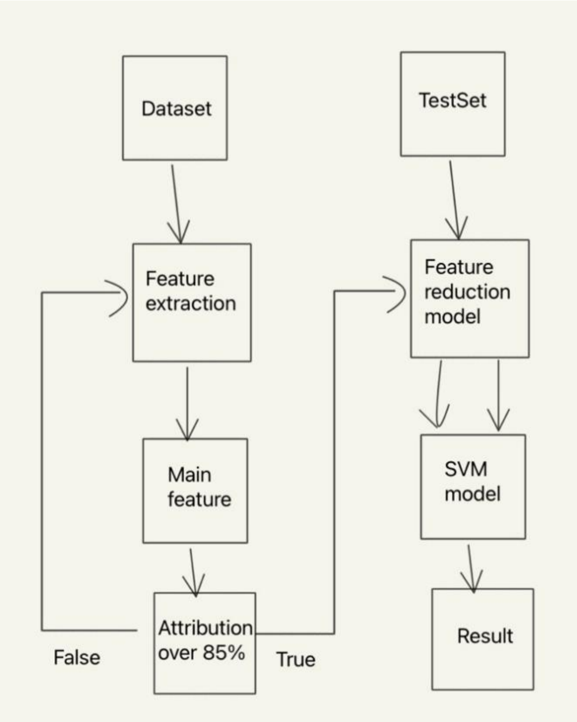

Figure 1. Flow chart.

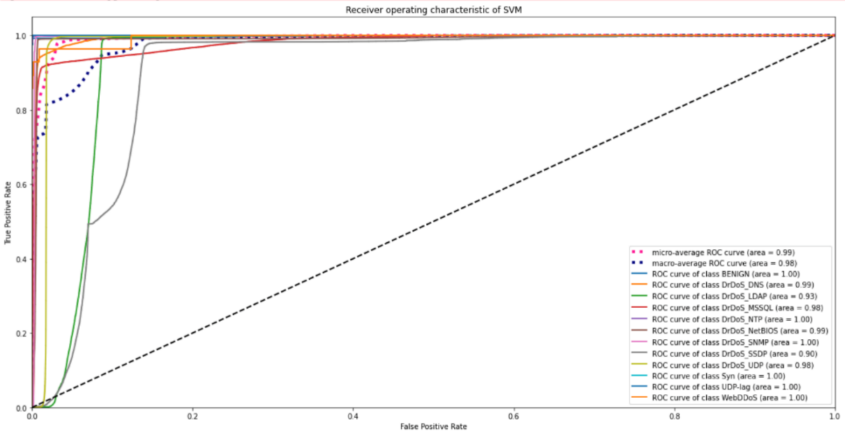

Figure 2. Transformation chart.

2.2. Random forest

Random Forest is an ensemble learning algorithm that is widely used in machine learning tasks such as classification and regression. The algorithm works by building multiple decision trees on a random subset of the training data and combining the outputs to make a final prediction. Random Forest is an improvement over single decision trees, as it reduces overfitting and improves prediction accuracy. As shown in Figure 2.

In this study, the Random Forest algorithm was implemented using the Random Forest Classifier from the scikit-learn package. The algorithm is trained on the DDoS dataset using 100 decision trees and the Gini criterion is used to measure the quality of the distribution. The minimum number of samples required for splitting nodes is set to 2 and the minimum number of samples required for leaf nodes is set to 1. As shown in Figure 3. This hyperparameter is set to prevent overfitting and achieve optimal performance on DDoS datasets.

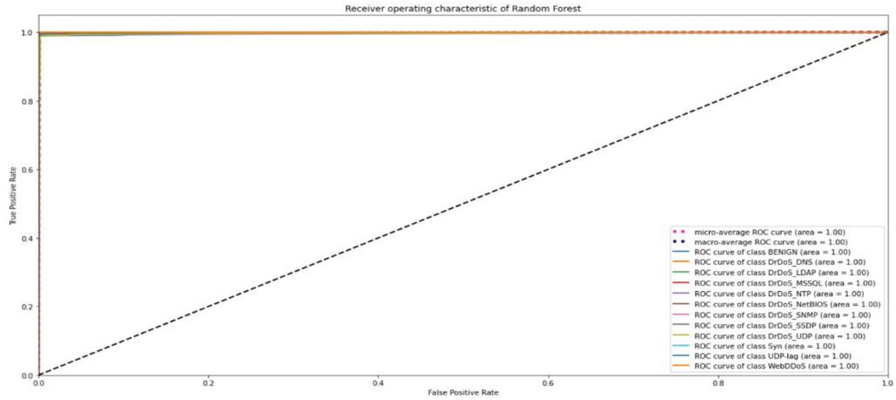

Figure 3. False positive rate.

Multi-Layer Perceptron (MLP) is a type of artificial neural network commonly used for classification and regression tasks. MLP consists of several layers of interconnected neurons that are regulated in a feedforward manner, meaning that the output of each neuron is used as input for the next layer. MLP is very effective at dealing with nonlinear relationships between input and output variables, and can be used to study complex patterns in data.

In this research, the MLP algorithm is implemented using the Keras library from TensorFlow. The model architecture consists of many layers of densely connected neurons, with the ReLU activation function used in the hidden layer and the Softmax activation function used in the output layer to achieve multiclass classification. The optimizer used is adadelta, and categorical crossentropy is used as the loss function. As shown in Figure 4. The model hyperparameters are tuned to prevent overfitting and achieve optimal performance on DDoS datasets.

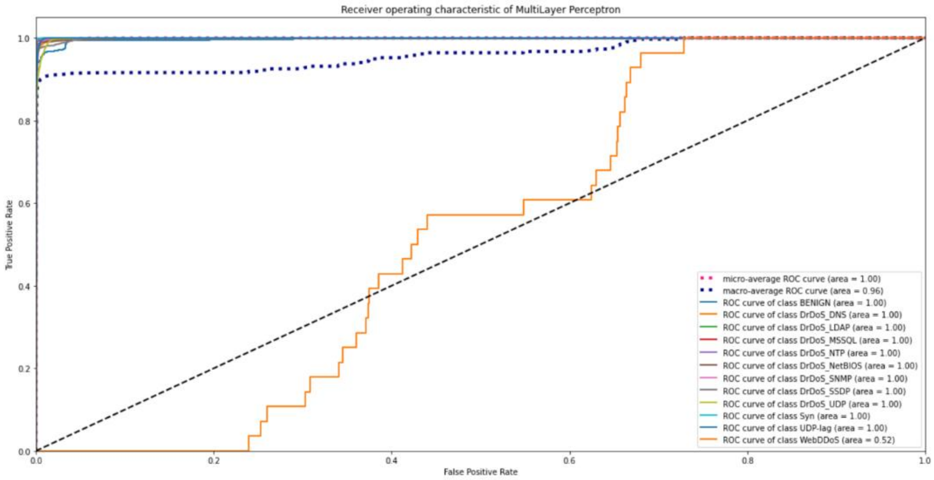

Figure 4. False positive rate.

3. Experiment and analysis

3.1. Dataset

The CICDDoS2019 dataset is specifically designed to be used for detecting DDoS attacks. This is because the dataset contains traffic data that was generated during a simulated DDoS attack on a web server. The features in the dataset are carefully selected to capture the characteristics of DDoS traffic, such as high packet rates, large flow durations, and high flow sizes.

Furthermore, the dataset contains a large number of samples, with over 2.5 million records, which makes it a valuable resource for training and evaluating machine learning models for DDoS detection.

Additionally, the dataset contains both normal and attack traffic, which provides a balanced distribution of data for training and testing purposes. As shown in Table 1.

Table 1. Experimental analysis.

Flow Duration	The time between the first and last packet of the flow (in milliseconds)
Total Fwd Packets	The total number of packets in front
Total Backward Packets	A11 packets are in the reverse di rection
Fwd Packet Length Max	Maxi imum length of packet in forward direction (in bytes)
Fwd Packet Length Min	Mini imum length of packet in forward direction (in bytes)
Fwd Packet Length Mean	Average length of packet in forward di rect ion (in bytes)
Bwd Packet Length Max	Max imum length of packet in reverse direction (in bytes)
Bwd Packet Length Min	Minimum length of packets in reverse direction (in bytes)
Bwd Packet Length Mean	Average length of packets in reverse direction (in bytes)
Flow Bytes/s	The number of bytes trans ferred per second in the flow (length size / flow length, in bytes per second)
Flow Packets/s	Number of packets sent per second in a flow (total packets in the flow/time flow, in packets per second)
Flow IAT Mean	The time between two consecutive packets sent from the same source (in milliseconds)
Flow IAT Std	Standard deviation of time between two successive packets sent by the same source _(in milliseconds)
Flow IAT Max	Max imum t ime between two consecutive packets sent by the same source (in milliseconds)
Flow IAT Min	Min imum time between two consecut ive packets sent by the same source (in milliseconds)
Fwd IAT Total	Total time between two successive packets in the forward direction (in milliseconds)
Fwd IAT Mean	Average t ime between two successive packets in the forward direction (in milliseconds)
Fwd IAT Std	Standard deviation of time between two successive packets in the forward direction (in milliseconds)
Flow Duration	The time between the first and last packet of the flow (in milliseconds)
Fwd IAT Max	Max: imum time between two successive packets in the forward direction (in milliseconds)
Fwd IAT Min	Minimum time between two successive packets in the forward direction (in milliseconds)

Table 1. (continued).

Bwd IAT Total	Total time between two successive packets in the backward direction (in milliseconds)
Bwd IAT Mean	Average time between two success ive packets in the reverse direction (in milliseconds)
Bwd IAT Std	Standard deviation of time between two successive packets in the backward direction (in milliseconds)
Bwd IAT Max	Max imum time between two successive packets in the reverse direction (in milliseconds)
Bwd IAT Min	Minimum time between two successive packets in the reverse direction (in milliseconds)
Label	Traffic flow class label, which can be either a normal flow or a DDoS attack flow

3.2. Experimental evaluation and result

The performance of each algorithm is evaluated using cross-validation to ensure its accuracy is not due to overfitting. Two evaluation methods are involved, accuracy score and F1 score. The accuracy of each algorithm is measured using metrics such as precision, recall, and F1-score. As shown in Table 2. Metrics are calculated for each class, and the overall performance of each algorithm is determined based on these metrics.

Table 2. Experimental result.

Algorithm	Accuracy	Precision	Recall	F1 Score
SVM	99.58%	99.58%	99.58%	99.58%
Random Forest	99.98%	99.99%	99.98%	99.98%
ML P	99.99%	99. 99%	99. 99%	99.99%

4. Conclusion

Based on the above metrics and the evaluation DDoS attack detection method based on decision tree, still have room for improvement in the detection rate and time consumption of this method and a DDoS attack detection methods in the SDN environment, SVM has good classification performance, but it has a large time consumption. K-nearest neighbor has high efficiency, but the need to calculate the distance between the test points and each data point in the training set leads to high space complexity. Which leads to an improved algorithm that combine the two algorithm RF-SVM. The basic idea is to introduce the information gain feature of random forest into the SVM algorithm, using information gain as the weight value of traffic features, screening out features with low importance, and improving the detection rate, recall rate, and F1 value of traffic classification. Compared to SVM, RF-SVM can handle high-dimensional data more efficiently by reducing the number of input features through Random Forest's feature selection. This can improve classification accuracy and reduce overfitting. Compared to RF, RF-SVM can achieve better performance in cases where the dataset has a limited number of training samples, which is a common problem in many real-world applications. RF-SVM can effectively reduce the number of input features and improve classification accuracy with fewer training samples. Compared to MLP, RF-SVM is less prone to overfitting, and the feature selection process in RF-SVM can help to reduce the impact of noise in the data. Additionally, RF-SVM can provide a better balance between accuracy and computational complexity in some cases. In summary, RF-SVM can be a better choice than SVM, RF, and MLP in certain cases where the dataset is high-dimensional, has a limited number of training samples, or requires a balance between accuracy and computational complexity. However, the best choice of algorithm depends on the specific problem and dataset, and it's important to evaluate different algorithms carefully to find the optimal solution.

References

[1] Liu, S., Sun, X., Wang, X., Tian, Y., & Yang, X. (2021). An efficient distributed detection framework against DDoS attacks in software-defined networks based on machine learning and blockchain. Future Generation Computer Systems, 116, 792-804.

[2] Lin, Y. T., Kuo, W. T., & Chang, H. S. (2021). Machine learning based techniques for DDoS attack detection on cloud computing. Cluster Computing, 24(3), 2257-2266.

[3] Eisa, M., Kim, S., Zhang, J., & Khan, S. U. (2021). A neural network-based system for improved detection and mitigation of DDoS attacks. Journal of Network and Computer Applications, 169, 102807.

[4] Babuka, R., He, F., & Kumar, R. (2020). Detection of DDoS attacks using machine learning: A review. Computer Science Review, 35, 100233.

[5] Kharde, V. B., & Khatri, S. K. (2020). Machine learning based detection of botnet based DDoS attack. Journal of Ambient Intelligence and Humanized Computing, 11(1), 401-412.

[6] Njogu, S., Lee, J. M., & Kim, J. M. (2021). A machine learning approach for DDoS attack detection in the cloud environment. Journal of Network and Computer Applications, 178, 102937.

[7] Yang, Z., Lv, X., Liu, X., & Guo, S. (2021). Anomaly detection of DDoS attacks using machine learning techniques considering time segmentation. Applied Sciences, 11(8), 3573.

[8] Tian, T., Gao, Y., Jacob, G., & Duan, Q. (2020). An approach for DDoS attack detection using machine learning on time-dependent features. Pervasive and Mobile Computing, 68, 101221.

[9] Tang, Y., Xiao, Z., Chen, L., & Fan, X. (2021). A multi-dimensional machine learning-based detection model against DDoS attacks for industrial control systems. IEEE Transactions on Industry Applications, 57(1), 160-170.

[10] Xi, H., Zhang, R., & Hu, B. (2020). A deep learning-based DDoS attack detection system using convolutional neural network and SVM. Journal of Ambient Intelligence and Humanized Computing, 11(11), 4331-4342.