A Machine Learning Based Classification and Prediction Technique for DDoS Attacks using KNN and Naïve Bayes Algorithm

Abhijith V Nair1
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

Mr.Eldhose K Paul5, Chandralekha j3
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

Adarsh P Baiju2
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

Govind G Das 4
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

***Computer Science &Engineering Mangalam Colleg Of Engineering Abstract*— A distributed denial of service (DDoS) attack isa subclass of a denial of service (DoS) attack. DDoS attacks involve multiple interconnected online devices, collectively known as botnets, that are used to attack targeted websites with bogus traffic. Unlike other types of cyberattacks, DDoS attacks do not attempt to breach your perimeter. Rather, it aims to prevent legitimate users from accessing her website and servers. DDoS can also beused as a pretext for various malicious sports, shutting down protective devices, and violating a target's protective perimeter. Classify and predict DDoS attacks using a machine learning approach. I used Naive Bayes and KNN algorithms. UNSW-nb15 record on GitHub 1 with functional data on DDoS attacks. Design a framework for DDoS attack classification and prediction based on existing datasets using machine learning techniques**

***Keywords—DDoS attacks, machine learning, random forest, XGBoost, prediction*.**

I. INTRODUCTION

Distributed Denial of Service (DDoS) attacks are a subclass of Denial of Service (DoS) attacks. DDoS attacks involving many interconnected online devices, collectively known as botnets, are used to attack targeted websites with spoofed traffic. Unlike other types of cyberattacks, DDoS attacks do not attempt to penetrate your perimeter. Instead, it aims to prevent legitimate users from accessing the website and its servers. DDoS can also be used as an excuse for various malicious activities, shutting down protection devices and breaching the target's perimeter. Classification and prediction of DDoS attacks using machine learning. I used Naive Bayes and KNN algorithms. Register UNSW-nb15 onGitHub 1 with functional data on DDoS attacks. Design a DDoS attack prediction and classification framework based on existing data sets using machine learning techniques for set of concern patterns to detect unusual behavior that could indicate a DDoS attack. Additionally, organizations can use intrusion detection and prevention systems (IDPS) to identify and block DDoS traffic. It is important to be able to predict and prevent DDoS attacks, as they can significantly disrupt business operations and lead to financial loss. DDoS attacks can also be used as a smokescreen to mask other malicious activity, such as data theft or system intrusion. A Distributed Denial of Service (DDoS) attack is a malicious attempt to disrupt the normal operation of a targeted website, server, or network. flooded it with a flood of traffic from multiple sources. This traffic stream can come from compromised devices that are part of a botnet, which is a network of devices controlled by the attacker.

The purpose of a DDoS attack is to make the targeted system inaccessible to legitimate users, thereby denying them access to the services provided by the system. This can cause significant damage to businesses and organizations that rely on their online presence for customer acquisition, sales, or communication. These attacks take advantage of specific limitations that apply to any arrangement asset, such as the framework of the autho- rized organization’s website. To predict a DDoS attack, one can use various techniques such as statistical analysis, machine learning, and artificial intelligence. These techniques can analyze network traffic patterns to detect anomalous behavior that could indicate a DDoS attack. Additionally, organizations can use intrusion detection and prevention systems (IDPS) to identify and block DDoS traffic. It is important to be able to predict and prevent DDoS attacks as they can significantly disrupt business operations and lead to financial loss. DDoS attacks can also be used as a smokescreen to mask other malicious activity, such as data theft or system intrusion. These attacks take advantage of specific limitations that apply to any arrangement, such as an authorized organization's web framework. A DDoS attack sends different requests (with IP spoofing) to the target web resources to overwhelm the website's ability to handle different requests,and cause website cannot function properly.

II. RELATED WORK

One way to use deep learning algorithms to predict DDoS attacks is to train a neural network to recognize regular network traffic patterns. The network can then be used to identify unusual traffic patterns that could indicate a DDoS attack. The more data available to train the network, the better the network can recognize patterns and detect anomalies. Another approach is to use deep learning algorithms to analyze log data from network devices and servers. This data can be used to train a model that can recognize traffic patterns and identify potential DDoS attacks. By analyzing log data in real time, deep learning algorithms can help detect DDoS attacks early and prevent them from causing significant damage.

[1] Adversary machine learning is applied to malware and intrusion scenarios: Enemy defenses have been significantly less explored, although they have also been shown to be effective against enemy attacks. We also conclude that, unlike malware scenarios, dataset diversity in intrusion scenarios is still very low, most used very outdated data set. In this article, we explored several works applying machine learning concepts that contradict malware detection and intrusion scenarios. We first covered various fundamental concepts that can help in understanding the basics of your opponent's machine learning, as well as your opponent's offensive and defensive strategies. We then explored the application of these techniques to intrusion detection and malware detection scenarios and concluded that adversarial attacks can deteriorate the performance of malware and intrusion classifiers, even if they follow different architectures or are from different families due to the transferability of adversarial attacks across different classifiers; Various adversarial attack strategies have been explored for both scenarios, with some strategies being more effective than others depending on the situation. Their effectiveness was proven, but there is a wider variety of defensive strategies that were proposed to image recognition that can be applied to intrusion and malware detection (also presented in section IV) as well as other deep learning based anomaly detection techniques .The usage of more recent and standardized datasets: In the case of intrusion detection, the main dataset used is NSL-KDD, which is greatly outdated, but is the mainly studied one due to lack of available alternatives.

Increasing the Performance of Machine Learning-Based IDSs on an Imbalanced and Up-to-Date Dataset: The used datasets were collected during a limited period in some specific networks and generally don't contain up-to-date data. Additionally, they are imbalanced and cannot hold sufficient data for all types of attacks. These imbalanced and outdated datasets decrease the efficiency of current IDSs, especially for rarely encountered attack types. In this paper, we propose six machine learning-based IDSs using K Nearest Neighbor, Random Forest, Gradient Boosting, Adaboost, Decision Tree and Linear Discriminant Analysis algorithms. For a more realistic IDS implementation, an up-to-date security dataset, CSE-CIC-IDS2018, was used instead of the older dataset and is already working for the most part. The selected data set is unbalanced Therefore, in order to increase the efficiency of system depending on the attack types and reduce the missed intrusions and false alarms, the imbalance rate is reduced by using an aggregate data generation model that can The name is Synthetic Minority Oversampling Technique (SMOTE). Data generation is done for small classes and their number is increased to average data size through this technique. The experimental results have demonstrated that the proposed method significantly increases the detection rate of rare intrusions. Experimental results show that the deployed models have very good accuracy compared to recent documents. Using the sampled data set resulted in an increase in the model's mean accuracy from 4.01% to 30.59%.

[2] Development of an online hate classifier for multiple social media platforms: Intrusion detection can identify unknown attacks from network traffic and is an effective means of network security. Today, existing methods for network anomaly detection are often based on traditional machine learning models, such as KNN, SVM, etc. While these methods can achieve out-of-the-box functionality, they achieverelatively low accuracy and rely heavily on manual traffic feature design. has become obsolete in the age of big data. To solve the low accuracy and feature engineering problems in intrusion detection, the BAT traffic anomaly detection model is proposed. Experimental results on the NSL-KDD dataset show that the BAT-MC model achieves high accuracy. Compared with several standard classifiers, these comparisons show that the results of BAT-MC models are very promising compared with other current methods based on deep learning. Therefore, we believe that the proposed method is a powerful tool for the intrusion detection problem.

[3] Network intrusion detection based on PSO-Xgboost model: Network Intrusion Detection System (NIDS) is a commonly used tool to detect attacks and protect the network, while one of its general limitations is the problem of false authentication. Based on our experience comparing and analyzing the characteristics of Particle Swarm Optimization (PSO) and Xgboost. overall classification accuracy is higher than other alternative models such as Xgboost, Random Forest, Bagging and Adaboost. First, a classification model based on Xgboost is built, then PSO is used to search the optimal structure of Xgboost adaptively. The NSL-KDD standard dataset was used to evaluate the proposed model. Our test results demonstrate that the PSO-Xgboost model outperforms other comparison models in terms of accuracy, recall, macro mean (macro) and mean accuracy. average (mAP), especially when identifying minority group attacks such as U2R and R2L. The proposed method gives an idea of the applications of swarm intelligence

in NIDS, which can also be applied to solve other classification problems. There are still some functions in this model to improve. If the number of elements or the number of iterations is small, the algorithm is likely to fall into a locally optimal solution.

[4] Similarity-based feature conversion for network anomaly detection: The basic goal of any network intrusion detection system is to automate the work The results of the tests demonstrated that our anomaly detection method applying the proposed feature conversion technique is relatively better than

the CANN, GARUDA and UTTAMA detection methods discussed in recent research papers. In this paper, we have applied the proposed distance function to perform feature clustering and perform feature transformation. Hence, the dimensionality reduction is done through feature transformation. Automated detection systems designed to identify anomalous inbound traffic patterns typically employ widely used machine learning techniques. However, regardless of the system model developed to identify anomalous traffic, all of these models require a comparison of abnormal and normal traffic patterns. The distance function proposed in this work is designed by considering the basic Gaussian membership function. After performing dimensionality reduction using the proposed feature extraction technique, we applied the classification algorithms to evaluate the performance of the classifiers on the transformed dataset.

III. METHODOLOGY

A. *Proposed System* III

we design a framework for the DDoS attack classification and prediction based on the existing dataset that used machine learning methods. This framework involves the following main steps.

1) The first step involves the selection of dataset for utilization.
2) The second step involves the selection of tools and language.
3) The third step involves data pre-processing techniques to handle irrelevant data from the dataset.
4) In the fourth step feature extraction and label.
5) In the fifth step, the data splitting is performed into a train and test set for the model. In this step, we build and train our proposed model.

B. Algorithm

1. Naïve bayes algorithm

Here are the steps to implement Naive Bayes algorithm for DDoS attack prediction:

Collect the training data: The first step is to collect a large dataset of network traffic containing both normal and attack traffic. This dataset will be used to train the Naive Bayes algorithm.Collect the training data ,The first step is to collect a large dataset of network traffic containing both normal and attack traffic. Dataset will be used to train the Naive Bayes algorithm.

1. Feature extraction: The next step is to extract relevant features from each packet in the dataset. Some of the features that can be used for DDoS attack prediction include packet size, packet rate, protocol, source IP address, destination IP address, and so on.

1. Data preprocessing: Once the features have been extracted, the next step is to preprocess the data. This can involve normalizing the features, converting categorical variables to numerical variables, and so on step 4) Training the Naive Bayes algorithm: Once the data has been preprocessed, the next step is to train the Naive Bayes algorithm using the training data.
2. Testing the Naive Bayes algorithm: After training, the Naive Bayes algorithm is tested on a separate dataset containing both normal and attack traffic. The algorithm predicts whether each packet is normal or an attack based on its features.
3. Evaluating the performance: The final step is to evaluate the performance of the Naive Bayes algorithm. This can involve calculating metrics such as accuracy, precision, recall, and F1 score.

2. K-Nearest Neighbor (KNN) Algorithm

Here are the steps to use KNN for DDoS attack prediction and classification:

1. Data Preprocessing: The first step is to preprocess the data by cleaning, normalizing, and transforming it into a suitable format for the algorithm.
2. Feature Selection: The next step is to select the relevant features from the dataset. These features should be the ones that are most relevant to the classification task.
3. Splitting the Data: Split the data into a training set and a test set. The training set will be used to train the KNN algorithm, while the test set will be used to evaluate the performance of the algorithm.
4. Choosing K: Choose the value of K, which is the number of nearest neighbors to consider in the classification process. This value should be chosen based on the dataset and the problem being solved.
5. Training the Algorithm: Train the KNN algorithm on the training set. The algorithm will calculate the distance between each data point and all other data points in the training set.
6. Predicting the Class: For each data point in the test set, the KNN algorithm will find the K nearest neighbors and assign the majority class among them as the predicted class for that data point.
7. Evaluating the Performance: Evaluate the performance of the KNN algorithm using metrics such as accuracy, precision, recall, and F1 score. These metrics will help determine the effectiveness of the algorithm in predicting and classifying DDoS attacks.

In summary, KNN is a useful algorithm for DDoS attack prediction and classification. With proper data preprocessing, feature selection, and parameter tuning, KNN can accurately classify network traffic as normal or attack traffic. KNN is simple to understand and implement, making it a popular choice for many machine learning applications.

KNN can handle noisy data and can easily adapt to changes in the data. Naive Bayes is computationally efficient and can handle large amounts of data. Naive Bayes can be trained

C. System Architecture

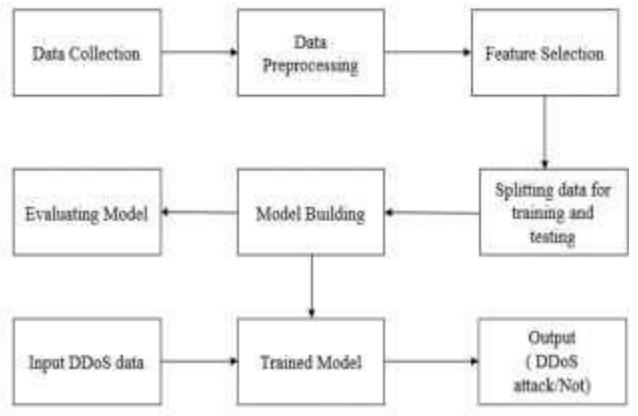

fig 1: system architecture

Data flow diagram

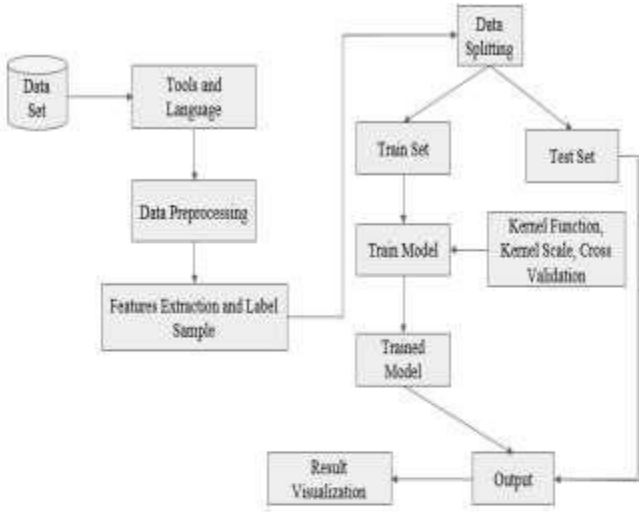

fig 2: data flow diagram

IV. RESULT

KNN (K-Nearest Neighbors) algorithm is a non- parametric and instance-based machine learning algorithm that is often used for classification problems. The algorithm works by finding the K nearest instances to the input instance in the training data, and then predicting the class of the input instance based on the majority class among the K nearest instances. KNN does not make any assumptions about the underlying data distribution, and it can work well with noisy and complex data.

Naive Bayes is a probabilistic machine learning algorithm that is based on the Bayes theorem. It is often used for classification problems, and it assumes that the features of the input instance are conditionally independent given the

quickly, making it useful for real-time DDoS attack prediction. Naive Bayes is less prone to overfitting, meaning that it is less likely to make mistakes when predicting DDoS attacks based on noisy data.

class label. Naive Bayes calculates the posterior probability of each class given the input features, and then predicts the class with the highest probability.

Naïve Bayes is a simple and fast algorithm that can work well with high-dimensional data. To evaluate the performance of KNN and Naive Bayes algorithms for DDoS attack classification and prediction, we need to collect a dataset of labeled instances of DDoS attacks. The dataset should include features such as source IP address, destination IP address, protocol, port number, packet size, packet rate, etc. The dataset should also include labels indicating whether each instance is a DDoS attack or not. Once we have a dataset, we can split it into training and testing sets, and then train KNN and Naive Bayes algorithmson the training set. We can then evaluate the performance of the algorithms on the testing set by calculating metrics such as accuracy, precision, recall, F1-score, etc. The results of the classification and prediction using KNN and Naive Bayes algorithms will depend on the quality of the dataset and the choice of hyper parameters for each algorithm. It is possible that one algorithm may perform better than the other for certain types of DDoS attacks, or for certain features of the input instances. Therefore, it is important to try multiple algorithms and compare their performance on the same dataset

VI. CONCLUSION

A complete systematic approach for detection of the DDoS attack we got Improved Network Security, Faster Response Time, Enhanced Risk Management etc.. First, we selected the UNSW-nb15 dataset from the GitHub repository that contains information about the DDoS attacks. Python and jupyter notebook were used to workon data wrangling .we normalized the dataset for the algorithm. After data normalization, we applied the proposed , supervised, machine learning approach. The model generated prediction and classification outcomesfrom the supervised algorithm. Then, we used Naïve Bayes and KNN classification algorithms.By comparing the proposal to existing research works, the defectdetermination accuracy of the existing research which was 90% and 92% were also significantly improved.

VII. ACKNOWLEDGEMENT

The authors would like to thank Principal Vinodh P Vijayan, Neethu Mariya John, H.O.D, Faculty of Computer Science, for their appropriate guidance, valuable assistance and helpful comments during the proofreading process.

REFERENCES

[1] A machine learning based classification and prediction technique for ddos attacks ismail, muhammad ismail mohmand , hameed hussain, ayaz ali khan ubaid ullah [1], muhammad zakarya

, (senior member, ieee), aftab ahmed mushtaq raza , izaz ur rahman , and muhammad Haleem G. Karatas, O. Demir, and O. K. Sahingoz, ‘‘Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset,’’ *IEEE Access*, vol. 8, pp. 32150–32162, 2020.

[2] T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, ‘‘BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset,’’ *IEEE Access*, vol. 8, pp. 29575– 29585, 2020.

[3] H. Jiang, Z. He, G. Ye, and H. Zhang, ‘‘Network intrusion detection based on PSO-xgboost model,’’ *IEEE Access*, vol. 8, pp. 58392–58401, 2020.

[4] A. Nagaraja, U. Boregowda, K. Khatatneh, R. Vangipuram, R. Nuvvusetty, and V. S. Kiran, ‘‘Similarity based feature transformation for network anomaly detection,’’ *IEEE Access*, vol. 8, pp. 39184–39196, 2020.

[5] L. D’hooge, T. Wauters, B. Volckaert, and F. De Turck, ‘‘Classification hardness for supervised learners on 20 years of intrusion detection data,’’ *IEEE Access*, vol. 7, pp. 167455–167469, 2019.

[6] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, ‘‘An adaptive ensemble machine learning model for intrusion detection,’’ *IEEE Access*, vol. 7, pp. 82512–82521, 2019.

[7] Y. Yang, K. Zheng, B. Wu, Y. Yang, and X. Wang, ‘‘Network intrusion detection based on supervised adversarial variational auto-encoder with regularization,’’ *IEEE Access*, vol. 8, pp. 42169–42184, 2020.

[8] C. Liu, Y. Liu, Y. Yan, and J. Wang, ‘‘An intrusion detection model with hierarchical attention mechanism,’’ *IEEE Access*, vol. 8, pp. 67542–67554, 2020.

[9] S. U. Jan, S. Ahmed, V. Shakhov, and I. Koo, ‘‘Towarda lightweight intrusion detection system for the Internet of Things,’’ *IEEE Access*, vol. 7, pp. 42450–42471, 2019.

[5]