Detection Of Cyber Attacks in a Network Using Machine Learning Techniques

Madhan Kumar Jetty
Department of Computer Science and EngineeringBapatla Engineering College (Autonomous) (Affiliated to Acharya Nagarjuna University)
Bapatla, India
madhan.jetty@becbapatla.ac.in

Venkatesh Ravipudi
Department of Computer Science and Engineering
Bapatla Engineering College (Autonomous)
(Affiliated to Acharya Nagarjuna University)
Bapatla, India
venkateshravipudi555@gmail.com

Narendra Thammarapu
Department of Computer Scienec and Engineering
Bapatla Engineering College (Autonomous)
(Affiliated to Acharya Nagarjuna University)
Bapatla, India
narendrathammarapu9121@gmail.com

Shifa Anjum Shaik
Department of Computer Science and Engineering
Bapatla Engineering College (Autonomous)
(Affiliated to Acharya Nagarjuna University))
Bapatla,India
shifashaik2703@gmail.com

Sai Jagadeesh Siddabattuni
Department of Computer Science and Engineering
Bapatla Engineering College (Autonomous)
(Affiliated to Acharya Nagarjuna University)
Bapatla,India
saisj987@gmail.com

Abstract: In today's interconnected digital landscape, the threat of cyber attacks looms large, necessitating robust defense mechanisms. This project proposes a novel approach leveraging machine learning (ML) techniques for the detection of cyber attacks within network infrastructures. By harnessing the power of ML algorithms, including supervised, unsupervised, and deep learning methods, the system aims to identify anomalous patterns indicative of malicious activity. The dataset comprises network traffic data, enriched with labeled instances of known attacks for supervised learning, and unlabeled instances for unsupervised techniques. Through feature engineering, dimensionality reduction, and model optimization, the system can effectively discern between normal and malicious network behavior in real-time. The evaluation of the proposed methodology demonstrates promising results in terms of accuracy, precision, recall, and F1-score, highlighting its potential to enhance cybersecurity defenses and mitigate the impact of cyber threats.

Keywords: ids, machine learning, network security, xgboost, ensemble learning

I. Introduction

In today's interconnected world, the security of computer networks is of paramount importance. With the increasing prevalence of cyber threats and attacks, organizations and individuals alike face the challenge of safeguarding their networks from malicious actors. Traditional methods of network security, such as firewalls and intrusion detection systems (IDS), are no longer sufficient to defend against the evolving tactics employed by cybercriminals. As a result, there is a growing need for advanced techniques capable of detecting and mitigating network attacks in real-time.

Machine learning (ML) has emerged as a powerful tool in the field of network security, offering the potential to identify anomalous behavior and detect malicious activities with high accuracy. By leveraging large datasets and sophisticated algorithms, ML models can learn to distinguish between normal network traffic and suspicious patterns indicative of an attack. This capability makes ML-based intrusion detection systems (IDS) an attractive option for enhancing the security posture of organizations and minimizing the impact of cyber threats.

II. Related Work

Meftah et al., proposed anomaly-based NIDS with machine learning techniques. Random forest with 10-fold cross validation to assign the index of feature significance in reducing impurity in the whole forest. The top features of UNSW-NB15 Dataset are ct dst src ltm, ct srv dst, ct dst sport ltm, ct src dport ltm, ct srv src. Support vector machine with an accuracy of 82.11% outperformed Logistic Regression and Gradient Boost Machine in binary classification model for attack detection. For identifying the type of attack, the multi-classification model with Decision Tree C5.0, outperformed Naive Bayes and Support vector machine [5].

Machine learning models use content-based email filtering, which identifies some keywords that can produce high variance between spam and legitimate emails [1]. Malicious HTML code penetrations have many consequences, like disclosure of cookies, thereby altering the victim's page content.

The primary functions of NIDS are packet sniffing, identifying attack signatures, identifying attacks, and reporting attack details. Attacks are identified by capturing features from source and destination IP addresses, ports, protocol details, header details, etc. Based on the nature of attacks, attacks can be classified as passive and active [2].

Peng et al., proposed Deep Neural Network(DNN)k with five hidden layers to identify attacks (Normal, DoS, Probe Categories, R2L, U2R) with NSL-KDD Dataset and compared the performance with Machine Learning models (Support Vector Machines, Random Forest, Linear Regression Models). DNN produced satisfactory results for identifying Normal, Dos, and Prob categories. SVM performed well in detecting Normal and four attacks. Random forest and linear regression also performed well in identifying network attacks [3].

Attacks on computer networks are devastating and can affect the functioning of the entire system by reading, damaging, and stealing the data [4].

III. Overview

This study utilizes a dataset containing nearly 30 lakh records of 14 different attacks and genuine network data. Our main goal is to create a detection

model specifically designed to identify attacks performed in a network. In essence, the detection model operates as follows:

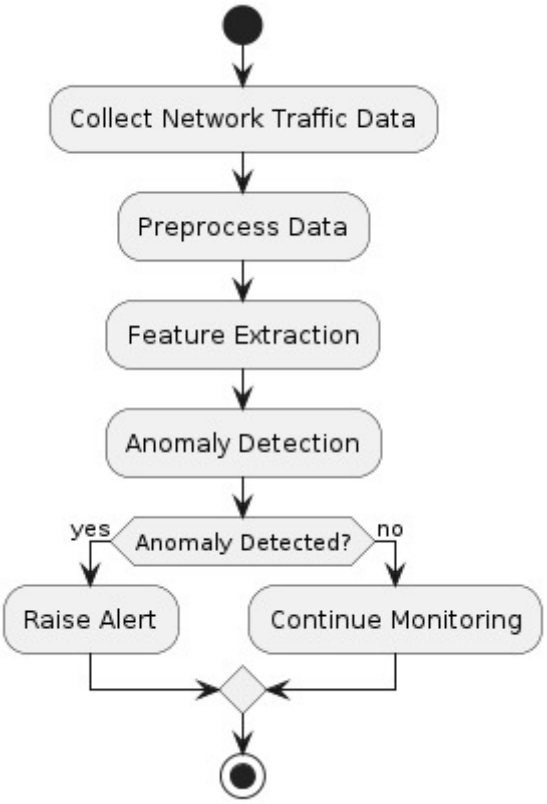

Fig.1 Proposed workflow of intrusion detection system

A. Input

The input for this study is taken from the CICIDS2017 dataset prepared by Canadian Institute for Cyber-Security(CIC) by capturing flow packets from a network over a period of one week

B. Data Pre-processing

To enhance data interpretability, several pre-processing steps are undertaken:

1. Importing libraries
2. Load Dataset into a DataFrame
3. Delete Duplicate Records
4. Remove Columns with only one value
5. Feature Selection
6. Splitting the dataset

C. Algorithms used:

The study employs the following algorithms:

1. XGBoost
2. Random Forest
3. Decision Tree
4. MLP
5. Gradient Boost

D. The developed system is trained to identify anomalous packets in the network thereby identifying any attack performed.

IV. METHODOLOGY

A. Data pre-processing

Data pre-processing involves transforming raw data into a usable format by applying various techniques. In this study, the following pre-processing steps were employed:

1. Importing Essential Libraries:

We began our project by importing the necessary Python libraries to facilitate data manipulation, analysis, and machine learning model implementation.

2. Loading Dataset into a DataFrame:

The next step involved loading the dataset into a DataFrame, a tabular data structure provided by the Pandas library. This allowed us to organize and manipulate the data efficiently.

3. Deleting Duplicate Records:

To ensure data integrity and accuracy, we identified and removed duplicate records from the dataset. This step helps prevent redundancy and maintains the quality of the data.

4. Removing Columns with Only One Value:

Subsequently, we inspected the dataset for columns containing only a single value across all records. Such columns provide no meaningful information for analysis and were thus removed to streamline the dataset.

5. Feature Selection:

Feature selection is a crucial step in preparing the dataset for model training. We carefully evaluated the relevance and importance of each feature and selected a subset of informative features to include in our analysis.

6. Splitting the Dataset:

Finally, we split the dataset into training and testing sets. This separation allows us to train our machine learning models on one portion of the data and evaluate their performance on another, independent portion. By doing so, we can assess the generalization capability of our models and ensure their reliability in real-world scenarios.

B. XGBoost

- XGBoost stands for Extreme Gradient Boosting, a powerful machine learning algorithm known for its efficiency and effectiveness in handling structured data.
- It belongs to the family of gradient boosting algorithms, which iteratively builds an ensemble of weak learners (decision trees) and combines them to form a strong predictor.
- XGBoost uses a gradient boosting framework, which optimizes the loss function by minimizing the gradient of the loss function with respect to the model parameters.
- It incorporates regularization techniques to prevent overfitting and improve generalization performance.
- XGBoost is widely used in various machine learning competitions and has become a popular choice for data scientists due to its speed and accuracy.

C. Random Forest

- Random Forest is an ensemble learning method that operates by constructing a multitude of decision trees during training.
- Each tree in the random forest is trained on a random subset of the training data and a random subset of the features.
- During prediction, the output of each individual tree is averaged or aggregated to obtain the final

prediction.

- Random Forest is robust to overfitting and noise in the data, making it suitable for a wide range of classification and regression tasks.
- It is known for its ease of use, scalability, and ability to handle high-dimensional data with large numbers of features.

D. Decision Tree

- A Decision Tree is a simple and intuitive machine learning algorithm used for both classification and regression tasks.
- It recursively partitions the feature space into disjoint regions, with each split maximizing the information gain or minimizing impurity.
- Decision trees are easy to interpret and visualize, making them useful for gaining insights into the data and understanding the decision-making process.
- However, decision trees are prone to overfitting, especially when the depth of the tree is not properly controlled.
- Techniques such as pruning, limiting the maximum depth, and using ensemble methods like Random Forest can help mitigate overfitting and improve performance.

E. Multi Layer Perceptron

- MLP is a type of artificial neural network composed of multiple layers of interconnected neurons.
- It consists of an input layer, one or more hidden layers, and an output layer.
- Each neuron in the network applies a weighted sum of its inputs, followed by a non-linear activation function, to produce the output.
- MLPs are capable of learning complex patterns and relationships in the data, making them suitable for a wide range of tasks, including classification, regression, and pattern recognition.
- Training an MLP involves iteratively adjusting the weights of the connections between neurons using optimization techniques such as gradient descent.

F. Gradient Boost

- Gradient Boosting is an ensemble learning technique that combines multiple weak learners (typically decision trees) to create a strong learner.
- Unlike traditional gradient descent, which updates the entire model simultaneously, gradient boosting builds the model sequentially by adding weak learners that complement the existing ones.
- Each new weak learner is trained on the residual errors of the previous ensemble, with the goal of reducing the overall error of the model.
- Gradient Boosting algorithms, such as Gradient Boosting Machines (GBM) and XGBoost, are widely used for regression and classification tasks due to their flexibility, robustness, and ability to handle heterogeneous data.

V. RESULTS

The comparative results after using the same dataset on above machine learning models are as shown below.

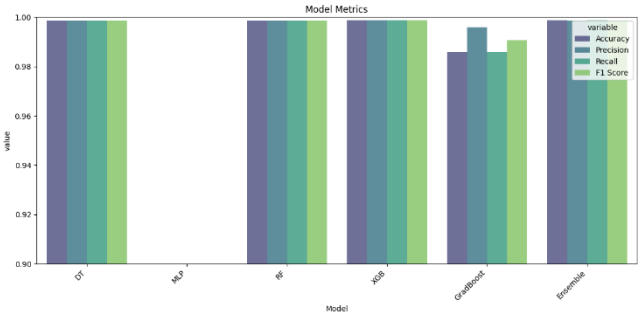

Fig. 2 Metrics for models

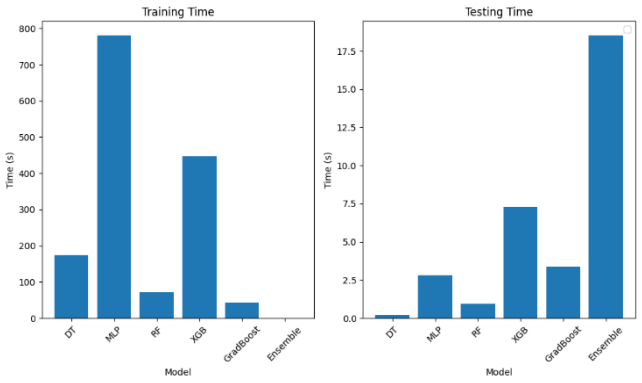

Fig. 9 Time for training and testing

From the above plots of results, it can be inferred that:

The custom RF, XGboost and DT model achieves an accuracy of 99.8, demonstrating its robustness.

Model	Accuracy	Precision	Recall	F1 Score
DT	0.9986	0.9986	0.9986	0.9986
MLP	0.8841	0.8640	0.8841	0.8726
RF	0.9985	0.9985	0.9985	0.9985
XGB	0.9987	0.9987	0.9987	0.9987
GradBoost	0.9858	0.9960	0.9858	0.9907
Ensemble	0.9987	0.9986	0.9987	0.9986

Table.1 Comparison of different models with their metrics

VI. CONCLUSION

In conclusion, XGBoost emerged as the top-performing model, closely followed by the ensemble model and random forest. These models demonstrated superior predictive performance and robustness, making them suitable candidates for deployment in practical applications. Further experimentation and fine-tuning may lead to even better results and provide valuable insights into the dataset.

VII. ACKNOWLEDGEMENT

The authors would like to thank Ass. Prof. Mr. J. Madhan Kumar, Bapatla Engineering College, Bapatla, for guiding throughout the work and the authors would also thank research paper writers for providing base to our work.

VIII. REFERENCES

[1] Dada, E.G., Bassi, J.S., Chiroma, H., Adetunmbi, A.O., Ajibuwa, O.E., et al., 2019. Machine learning for email spam filtering: review, approaches and open research problems. Heliyon 5, e01802.

[2] Garuba, M., Liu, C., Fraites, D., 2008. Intrusion techniques: Comparative study of network intrusion detection systems, in: Fifth International Conference on Information Technology: New Generations (itng 2008), IEEE. pp. 592–598.

[3] Peng, Y., Su, J., Shi, X., Zhao, B., 2019. Evaluating deep learning based network intrusion detection system in adversarial environment, in: 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), IEEE. pp. 61–66.

[4] Gandhi, M., Srivatsa, S., 2008. Detecting and preventing attacks using network intrusion detection systems. International Journal of Computer Science and Security 2, 49–60.

[5] Meftah, S., Rachidi, T., Assem, N., 2019. Network based intrusion detection using the unsw-nb15 dataset. International Journal of Computing and Digital Systems 8, 478–487.