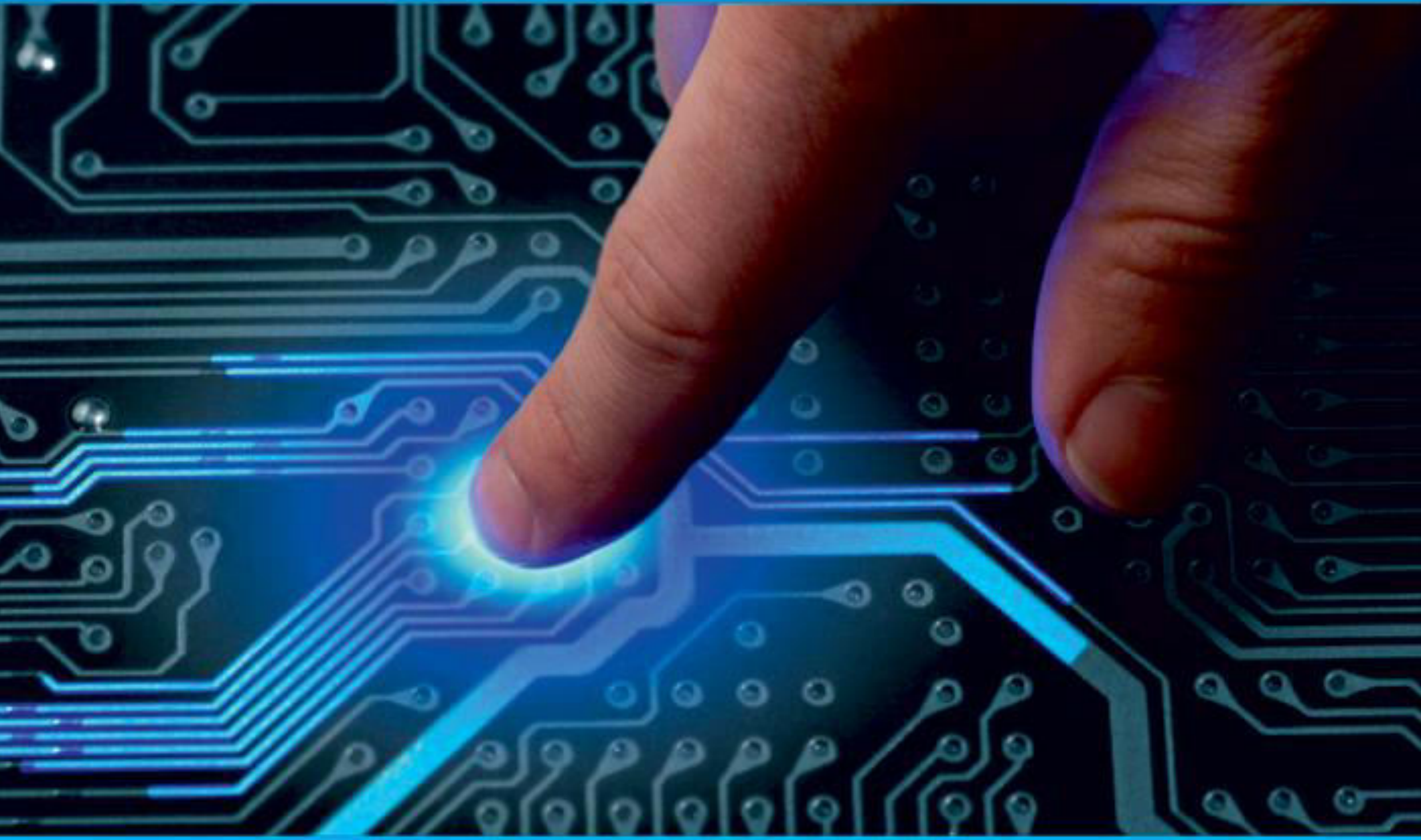
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 5, May 2022

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.165

9940 572 462 6381 907 438 ijircce@gmail.com www.ijircce.com

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005295 |

Analysis of Network Intrusion Detection using Feature Engineering in Ml

Sanka Sri Naga Sai Pravallika[1], Talluri Nandini[2], Pinnamsetty Lakshmi Sahithi[3], Satish Kumar Parasa[4]

UG Student, Dept. of I.T., Vasireddy Venkatadri Institute of Technology, Guntur, India[1,2,3]

Asst Professor, Dept. of I.T., Vasireddy Venkatadri Institute of Technology, Guntur, India[4]

ABSTRACT:Intrusion detection is one of the key interests in network administration and security. There is a need to protect the networks from known vulnerabilities and at the same time take steps to identify new and unknown but potential device abuses by creating more robust and effective systems for intrusion detection. An intrusion detection system (IDS) audits the traffic flowing in the network for suspicious activity and signals when any malicious activity is discovered. This type of IDS has the functionality to trace previously recognized patterns of pernicious activity going on in the network and spot intrusions.Our objective is to deploy a network based IDS that is programmed to detect any misuse of the network resources that it will detect malicious packets following in a network using feature engineering technique .We are going to analyze the network intrusion detection system with an algorithm that will give the more accuracy to the system is achieved.

KEYWORDS:Intrusion Detection, Data Set, Network, Feature Engineering, Accuracy.

I. INTRODUCTION

Intrusion Detection Systems (IDSs) have evolved into essential components of any computer network. An intrusion detection system (IDS) is a hardware or software system that monitors a company's computer network for potential threats or attacks. An IDS, on the other hand, is capable of reacting to any malicious transactions and reporting them to the appropriate authorities.Machine Learning (ML) based IDSs have emerged as the most leading systems in the intrusion detection research sector in previous years. Systems will be able to learn and improve utilizing prior data thanks to machine learning. As a result, ML-based computer programmers will not require explicit engineering (programmed). They are capable of self-learning and unsupervised machine learning. In supervised machine learning, models learn from data that has already been labeled. The data used to train models in unsupervised machine learning is unstructured (unlabeled).

This research focuses on supervised machine learning approaches, notably binary and multiclass classification tasks. When a supervised ML model is used to predict a discrete value, the classification procedure takes place. And in this scenario, the dataset used to train the models is typically large and has a high dimensional feature space. Because of this complexity, training and testing supervised machine learning models can take a long time. As a result, it's critical to use feature engineering techniques to reduce the number of features that aren't needed throughout the training and testing phases. In contrast to the wrapper-based feature extraction method. The space feature reduction process will be performed without the use of the classifier used for the final predictions using the filter-inspired feature extraction technique. In addition, the results obtained by various machine learning methods were surveyed and compared to those obtained in this project.

II. PROPOSED SYSTEM

The suggested model has the potential to detect malicious activities efficiently. Our main aim was to decrease false positive rates and decrease the time required to detect the intrusion. Therefore we used a feature engineering concept in our system , where the top 13 features are selected based on their information gain , and these features will be taken to build the ML model and the results will be predicted using this model. It also reduces complexity of the system , because it reduces the size of the data that will be fed into the ML model. Therefore time and complexity both can be reduced. We used the NSL-KDD dataset. According to our results we were able to conclude that our approach of using feature engineering to reduce the no of features and those will be used in the further process, has performed exceptionally well. We also compared three different classifiers , decision tree, Gaussian NB, Random Forest classifier and compared their accuracies , found out that Random forest had the highest accuracy which has been used for the rest of the process. And also a cross validation report is also displayed to show the working of our model.

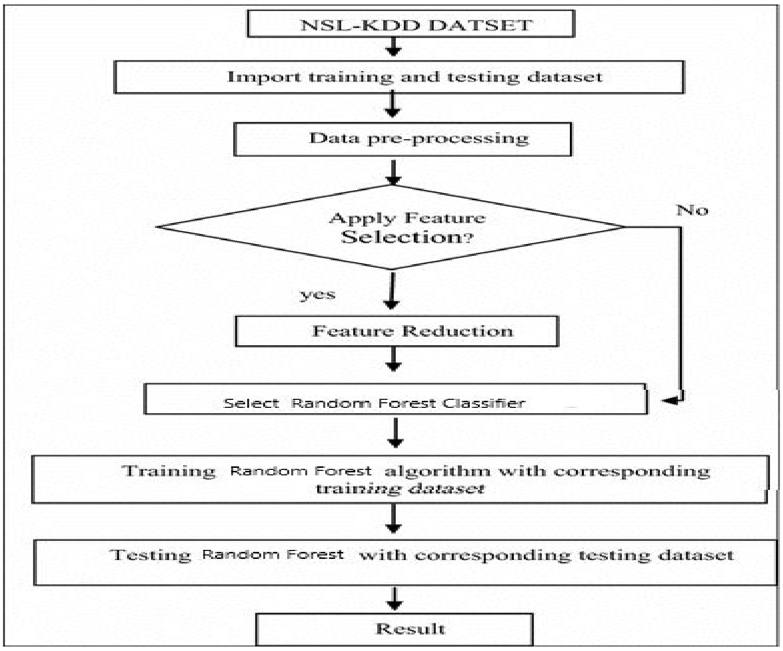

NSL-KDD DATSET
Import training and testing dataset
Data pre-processing
Apply Feature Selection?
No
yes
Feature Reduction
Select Random Forest Classifier
Training Random Forest algorithm with corresponding training dataset
Testing Random Forest with corresponding testing dataset
Result

Fig1: System architecture of proposed system

The advantages of Proposed system are :

- By using Feature engineering , the machine learning model can speed up the entire process.
- Reduces complexity of the system .
- It provides more accuracy.

III. METHODOLOGY

- **Dependencies:**

- **Numpy :**NumPy is a Python library used for working with arrays. It is an open source project and you can use it freely.
- **Pandas** : pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive.
- **Sklearn:** The sk learn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression etc.
- **Sys :**The sys module in Python provides various functions and variables that are used to manipulate different parts of the Python runtime environment.

- **Data pre-processing :**The process of converting raw data into a comprehensible format is known as data preprocessing. We can't work with raw data, so this is a key stage in data mining. This stage involves detecting categorical features, data encoding and data transformation.
- **Feature Selection :** We employed the concept of Recursive Feature Elimination(RFE). RFE is a

wrapper-type feature selection algorithm. This means that a special machine learning algorithm is given and utilized in the core of the tactic , is wrapped by RFE, and want to help select features. This process is repeated until a specified number of features remains. Next, we will evaluate an RFE feature selection algorithm on this dataset.

- **Build the model :**Random forest models are built. Random Forest could also be a well-liked machine learning algorithm that belongs to the supervised learning technique. It are often used for both Classification and Regression problems in ML. it's supported the concept of ensemble learning, which can be a process of blending multiple classifiers to unravel a complicated problem and to reinforce the performance of the model. It takes less training time as compared to other algorithms. It predicts output with high accuracy, even for the massive data set it runs efficiently.

IV. RESULTS & DISCUSSION

The proposed model was tested by applying the classificationaccuracy,confusion matrix, true positive, false positive, precision, recall, and f-measure on NSL-KDD Cup 99 dataset.

PerformanceMeasure:

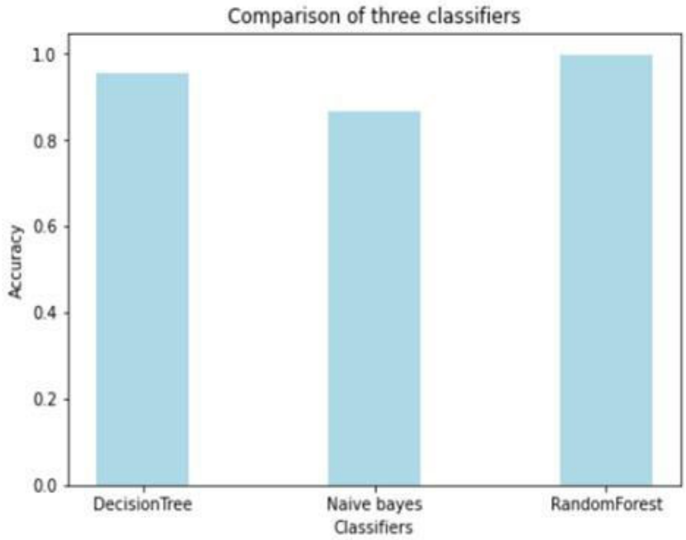
- Cross-Validation - we calculate accuracy, precision, recall and F-measure of the model.
- **Accuracy :** The accuracy (AC) is defined as the distribution of the total number of correct predictions. The equation is estimated as:
  Accuracy= TP+TN/TP+TN+FP+FN

- **Precision:** Precision is defined as the ratio of the total no. of true positives and the sum of the number of true positives and the number of false positives.
  It is calculated by the equation:
  Precision = TP/TP+FP

- **Recall:** Recall can be defined as the proportion of the total number of positive examples rightly listed, divided by the total number of positive ones. High Recall suggests correct identification of class. It is also defined in scientific terms as Detection Frequency, True Positive Rate, or Sensitivity. The equation computes as:
  Recall = TP/TP+FN

- **F-score:** The F score is also known as F1 score or F measure. It is a measure of the accuracy of a test. The F score is interpreted as the weighted harmonic mean of the test's precision and recall. This score is calculated using the following equation:
  F-Score = 2 * (precision * recall/ precision+recall)

  The analysis of the classifier will be done using evaluations techniques, cross validation, where confusion matrices for each type of attack will be calculated and displayed and cross validation will also be done to find out precision, Accuracy, Recall, F-measure will be calculated and displayed



V. CONCLUSIONS & FUTUREWORK

Conclusion:
Nowadays , intrusion detection systems are very important because every system is prone to attacks. We need to increase security in our daily use systems, which is why intrusion detection systems are very important. It helps us detect malicious activities efficiently. Existing intrusion detection systems detect anomalies and detect high false positive rates which leads to increase in rate of false alarms. Our main aim was to decrease false positive rates and decrease the time required to detect the intrusion. Therefore we used a feature engineering concept in our system , where the top 13 features are selected based on their information gain , and these features will be taken to build the ML model and the results will be predicted using this model. It also reduces complexity of the system , because it reduces the size of the data that will be fed into the ML model. Therefore time and complexity both can be reduced. We used the NSL-KDD dataset. According to our results we were able to conclude that our approach of using feature engineering to reduce the no: of features and those will be used in the further process, has performed exceptionally well. We also compared three different classifiers , decision tree, Gaussian NB, Random Forest classifier and compared their accuracies , found out that Random forest had the highest accuracy which has been used for the rest of the process. And also a cross validation report is also displayed to show the working of our model.

Future Work:
Proposing an effective Network Intrusion Detection System having an effective detection mechanism is a potential future scope of research in this area. For future research, we will use this knowledge to design a novel, lightweight, and efficient Network Intrusion Detection System which will effectively detect the intruders within the network. The research work can be extended by implementing various other learning Techniques.

REFERENCES

1. Maximilian Bachl, Joachim Fabini, and Tanja Zseby. 2021. A flow-based IDS using Machine Learning in eBPF. Cryptography and Security (cs.CR).Citation- [2102.09980v1] A flow-based IDS using Machine Learning in eBPF (arxiv.org)(Link to Research Paper)
2. Zeeshan Ahmad, Adnan Shahid Khan, Cheah Wai Shiang, Johari Abdullah, and Farhan Ahmad. 2021. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Trans. Emerg. Telecommun. Technol. 32, 1 (January 2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches - Ahmad - 2021 - Transactions on Emerging Telecommunications Technologies - Wiley Online Library (Link to Research Paper)
3. Khraisat, A., Gondal, I., Vamplew, P. et al. Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecure 2, 20 (2019). Survey of intrusion detection systems: techniques, datasets and challenges | Cybersecurity | Full Text (springeropen.com) (Link to Research Paper)
4. Gary Stein, Bing Chen, Annie S. Wu, and Kien A. Hua. 2005. Decision tree classifier for network intrusion

detection with GA-based feature selection. In Proceedings of the 43rd annual Southeast regional conference - Volume 2 (ACM-SE 43). Association for Computing Machinery, NewYork, NY, USA, 136–141. Decision tree classifier for network intrusion detection with GA-based feature selection | Proceedings of the 43rd annual Southeast regional conference - Volume 2 (acm.org) (Link to Research Paper)

5. NSL-KDD dataset NSL-KDD | Kaggle

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING