IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 12, Issue 1, December 2021
Impact Factor: 5.731

Anomaly Detection in Network Traffic Using Unsupervised Machine Learning Approach

Prof. Sagar Dhanake[1], Prathamesh Kulkarni[2], Himanshu Samariya[3], Akash Sitoke[4], Aman Chandre[5]
Assistant Professor, Department of Information Technology[1]
Student, Department of Information Technology[2,3,4,5]
Dr. D. Y. Patil Institute of Technology, Pune, Maharashtra, India

Abstract: *The advent of IoT technology and the increase in wireless networking devices has led to an enormous increase in network attacks from different sources. To maintain networks as safe and secure, the Intrusion Detection System (IDS) have become very critical. Intrusion Detection Systems (IDS) are designed to protect the network by identifying anomaly behaviors or improper uses. Intrusion Detection systems provide more meticulous security functionality than access control barriers by detecting attempted and successful attacks at the end-point or within the network. Intrusion prevention systems are the next logical step to this approach as they can take real-time action against breaches. To have an accurate IDS, detailed visibility is required into the network traffic. The intrusion detection system should be able to detect inside the network threats as well as access control breaches. IDS has been around for a very long time now. These traditional IDS were rules and signature-based. Though they were able to reduce false positives they were not able to detect new attacks. In today's world due to the growth of connectivity, attacks have increased at an exponential rate and it has become essential to use a data-driven approach to tackle these issues. In this paper, the KDD data set was used to train the unsupervised machine learning algorithm called Isolation Forest. The data set is highly imbalanced and contains various attacks such as DOS, Probe, U2R, R2L. Since this data set suffers from a redundancy of values and class imbalance, the data preprocessing will be performed first and also used unsupervised learning. For this network traffic based anomaly detection model isolation forest was used to detect outliers and probable attacks the results were evaluated using the anomaly score.*

Keywords: Anomaly Detection, Isolation Forest, Machine Learning, Intrusion Detection System, KDD Cup, NSL-KDD.

I. INTRODUCTION

As the number of networking devices are increasing at a very high rate in our day to day life and even at the workplace they handle very sensitive information. In recent years the number of unknown attacks have increased at an exponential rate so it is necessary to give secure network access to users and customers by keeping the network secure at the same time. The different ways to revert a digital attack is by utilizing Intrusion detection system (IDS). IDS is a system which keeps our data and information safe by keeping a check on the network for anomalies and any abnormal behavior. The IDS used in this study is anomaly-based. The outlier detection system assumes that any abnormal behavior is malicious. The main motive is to train the machine learning model with normal behavior and then search for the anomalies and raise alerts. There are different types of IDS , the initial models were not able to detect new attacks due to the increasing number of wireless devices and increase in the number of attacks.

II. BASIC TERMINOLOGIES USED

2.1 Isolation Forest

Isolation Forests (IF), similar to Random Forests, are build based on decision trees. And since there are no predefined labels here, it is an unsupervised model. Isolation Forests were built based on the fact that anomalies are the data points that are "few and different". In an Isolation Forest, randomly sub-sampled data is processed in a tree structure based on

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 12, Issue 1, December 2021
Impact Factor: 5.731

randomly selected features. The samples that travel deeper into the tree are less likely to be anomalies as they require more cuts to isolate them. Similarly, the samples which end up in shorter branches indicate anomalies as it was easier for the tree to separate them from other observations. To build an outlier detection model using isolation forest it is necessary to cautiously select n_parameters and contamination parameters.

III. Experimental Analysis

In this implementation system has been tried to detect anomalous points and there might be false positives but the main motive behind this was to detect maximum number of anomalous points or malicious attacks. The dataset used in this study is very huge and complex so it is impossible to test it on an ordinary processor. The experiment required performing data pre-processing and cleaning the data set. The experiment was performed in the ratio of 20% test and 80% training data. Isolation forest was used for identifying outliers . The algorithm used for this purpose is isolation forest. Various parameters and their values were used and tested to obtain a wide range of accuracy scores. Some attacks are more widespread than others . These attacks can make the model unfairly prejudiced. The main reason to use isolation forest is due to its better performance on highly imbalanced .Also, it is computationally more efficient. Training on smaller data sets and then reassembling datasets can be used.PCA was used to achieve dimensionality reduction and compression which helped to make the results of detection more conspicuous and easy to understand. The accuracy scores for each data point was plotted using decision function and analyzed.

3.1. Data Set

The NSL-KDD data-set contains 38 attacks which are combined into 4 basic attacks for better results. There were 38 different types of attacks, but only 24 are available in the training set. These attacks belong to four general categories: DOS, r2I, u2R, and prob attack. Given the descriptions of these attacks, one can observe that DOS attacks are different from other attacks. Dos attacks bring down a resource whereas these other ones intrude into a network or machine. However, all are equally dangerous and should be predicted with s ome accuracy. The majority of the samples are either DOS attacks or benign.

Description of the attacks in KDD data set as follows:

- **DOS:** This stands for Denial of service attack. It causes theserver to become unavailable by bombarding it with false requests. These attacks are very prevented in IoT and wireless networks. It can occur in the transport layer as well as the application layer.
- **Probe:** In this attack, the attacker sweeps through various hosts and services to identify open ports.
- **U2R:** User To Root attacks are less common as compared to Dos attacks. The attacker gains access and attempts to gain root privilege.
- **R2I:** It stands for Remote to local attack. In this attack, generally known to be propelled by an attacker to increase unapproved and remote access to a victim client machine in the whole system.

3.2 Data Pre-Processing

There are 41 features in the KDD99 dataset . 30 features contain continuous values. The data set contains integers as well as floating point numbers . Some of the character values require a certain amount of preprocessing, so they were assigned values in the form of binary numbers through one-hot encoding which converted the non-numeric values into numeric values which are given as the input to the algorithms. Some of the features had high ranging values as compared to others, which can affect the performance of the model severely. Thus; it has performed feature normalization and feature scaling using a standard scaler. Some feature values were categorical non-numeric data. Therefore, it also used Label Encoder () from the sklearn library to encode these values to put in our training process. It replaced the text data with new encoded values.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 12, Issue 1, December 2021
Impact Factor: 5.731

IV. System Design

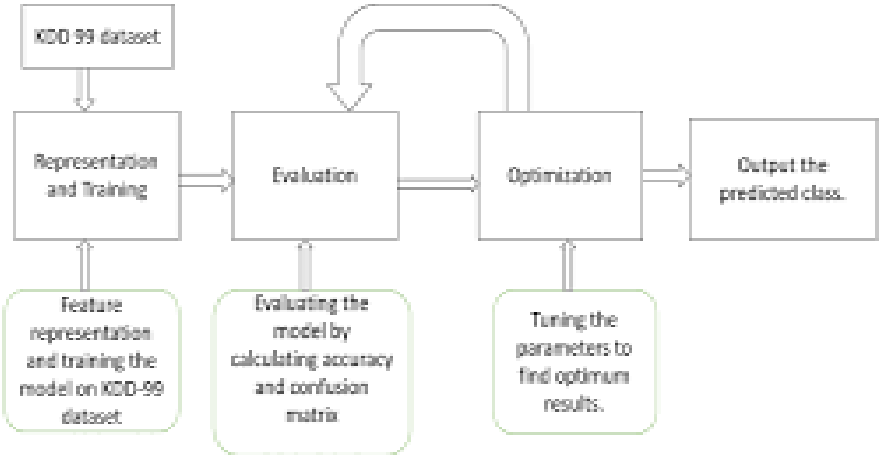


Figure 1: System Architecture

The isolation forest algorithm performs an unsupervised machine learning algorithm which builds random forests , which then analyzes the average depth required to isolate each point .There are various parameters used to build and instantiate the Isolation forest model. The most important parameter which doesn't affect the training of the model but is crucial in analyzing the output is the contamination parameter. Simply controls the threshold for the decision function when a scored data point should be considered an outlier. Time complexity is *O(no. of sample * n_estimators* log_sample_size*). It has linear complexity, which makes the isolation forest very efficient and suitable for real-time anomaly detection. A normal decision tree could overfit but due to forest ensemble technique the model does not suffer from overfitting. The following parameters were passed while creating and instantiating the Isolation forest object.

- *N_estimators:* determines how many trees/base estimators need to get built for estimation and detection of outlier class.
- *Max_sample:* samples determine how many training data points are picked to train each tree for analyzing.
- *Contamination_param:* It represents the proportion of outliers present in the given data set. The threshold for data points to be considered as anomalous is decided by the contamination factor. This was chosen as 1% for our data set.

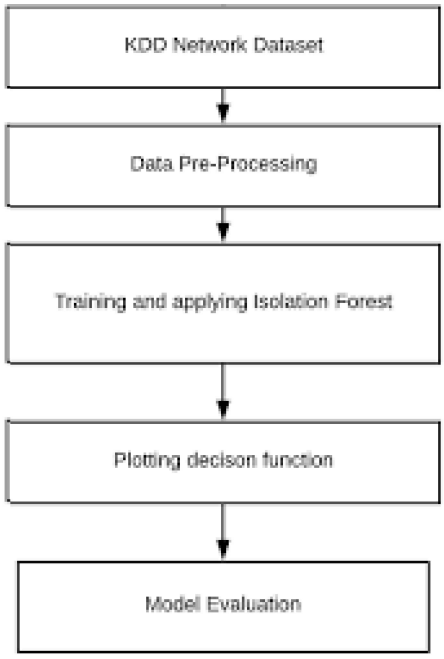
Since the model is unsupervised it needs to learn the presence of anomalous points by itself.

V. Algorithm/Flowchart

5.1 Training Models

The training and test split were performed to train the algorithms on training separately and testing on unknown examples to get better results. A portion of the data set was used to get better performance. Since the isolation forest is unsupervised as it doesn't require target labels when training the model. The various arguments required to initialize an Isolation forest model were analyzed and then supplied. Isolation forest is faster as compared to other algorithms.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 12, Issue 1, December 2021
Impact Factor: 5.731



VII. MATHEMATICAL MODEL

Since n/w traffic varies from weekdays and weekends it can have helper functions that indicate these % changes and display outliers based on

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}}$$

- when $E(h(x)) \rightarrow c(n), s \rightarrow 0.5;$
- when $E(h(x)) \rightarrow 0, s \rightarrow 1;$
- and when $E(h(x)) \rightarrow n - 1, s \rightarrow 0$

Where h(x) is the path length for the given data point after feature splits, c(n) is the average path length of unsuccessful search in a Binary Search Tree, and n is the number of external nodes. Each observation and subsequent feature value split is given an anomaly score and the following decision can be made on its basis: A score close to 1 indicates anomalies score much smaller than 0.5 indicates benign or normal points. If all scores are close to 0.5 then the entire sample does not seem to have distinct anomalies.

VIII. APPLICATIONS AND ADVANTAGES

Anomaly detection is applicable in a variety of domains such as :

- Intrusion detection,
- Fraud detection,
- Fault detection,
- System health monitoring,
- Event detection in sensor networks,
- Detecting ecosystem disturbances,
- Defect detection in images using machine vision.
- It is often used in preprocessing to remove anomalous data from the dataset.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 12, Issue 1, December 2021
Impact Factor: 5.731

IX. CONCLUSION

Unsupervised machine learning was used due to high imbalance in the data. The AUC score was computed 98.3%. The n_estimators parameter was kept at 100. The "contamination" parameter value was 4%.There is tremendous growth in the different types of network attacks and thus organizations are developing Intrusion Detection System (IDS) that are not only highly efficient but also capable of detecting threats in real time. It could be concluded that a more complete and clean data set leads to better results. The contamination parameter is very important in deciding the proportion of anomalies that could be detected. It is important to realize that machine learning, deep learning application is fairly new in the network security domain, and therefore there are still challenges related to scalability and efficiency.

ACKNOWLEDGEMENT

The completion of our project brings with it a sense of satisfaction, but it is never complete without those people who made it possible and whose constant support has crowned our efforts with success. One cannot even imagine our completion of the project without guidance and neither can we succeed without acknowledging it. It is a great pleasure that we acknowledge the enormous assistance and excellent co-operation to us by the respected personalities.

REFERENCES

[1]. G. Karatas et al., "Deep Learning in Intrusion Detection Systems" 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), Turkey, 2018.

[2]. H. Azwar et al., "Intrusion Detection in secure network for Cybersecurity systems using Machine Learning" 2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences ,Bangkok, Thailand, 2018.

[3]. Y. Chang et al., "Network Intrusion Detection Based on Random Forest and Support Vector Machine," IEEE International Conference on Computational Science and Engineering (CSE), Guangzhou, 2017.

[4]. Brao, Bobba et al., "Fast kNN Classifiers for Network Intrusion Detection System", Indian Journal of Science and Technology. 2017.

[5]. M. Z. Alom et all., "Network intrusion detection for cyber security using unsupervised deep learning approaches", 2017 IEEE National Aerospace and Electronics Conference (NAECON), Dayton, OH, 2017.

[6]. Mukkamala et al., "Intrusion detection using neural networks and support vector machines", International Joint Conference 2012.