INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

Machine learning algorithms for intrusion classification

Alapati Avinash
alapatiavinash.avinash@gmail.com
Anil Neerukonda Institute of Technology and Sciences, Bheemunipatnam, Andhra Pradesh

Alla Haripriya
aharipriya.15.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Bheemunipatnam, Andhra Pradesh

B. Sandeep
sandeepbali49@gmail.com
Anil Neerukonda Institute of Technology and Sciences, Bheemunipatnam, Andhra Pradesh

ABSTRACT

The usage of the Internet, the amount of network traffic volume is rapidly raising high. Preventing such huge, sensitive data from security attacks is equally important. Monitoring network traffic may indicate a possible intrusion in the network and therefore anomaly detection is important to detect and prevent such attacks. One of the approaches to anomaly detection is based on machine learning classification techniques. Here we apply seven different machine learning techniques: K-Means, K-Nearest Neighbours (KNN), Fuzzy C-Means (FCM), Support Vector Machine (SVM), Naïve-Bayes (NB), Radial Basis Function (RBF) and Ensemble method (Weighted average method) comprising of K-Nearest Neighbours and Naïve-Bayes (NB) on NSL-KDD dataset and evaluate the performance of these techniques. We also deduced how the change in the training size can affect the training time and the accuracy of each algorithm

Keywords— *Intrusion, Anomaly detection, NSL-KDD dataset*

1. INTRODUCTION

With networks more vulnerable and hackers equipped to cause havoc, it's no surprise that network attacks are on the rise. While firewalls and router-based packet filtering are necessary components for an overall network security topology, they are insufficient on their own [4]. Intrusion detection systems overcome such problems. They are effective when sophisticated attacks are embedded in familiar protocols. In the modern network, IDS has become an important and integral part of over-all security architecture. An IDS monitors and collects data from a target system (host or network), processes and correlates the gathered information, and initiates responses when evidence of an intrusion is detected. Responses from IDSs are usually reported to auto response systems or security staff for automatic or manual appropriate response actions. The machine learning algorithm is used in IDS because of its capability to classify normal/ attack network packets by learning from the collected data. The authors have conducted experiments to evaluate the best performing algorithm for Intrusion Detection and also deduce how the increase in train sample size can have an impact on train time and accuracy. To observe this all, experiments are carried out on various clustering and classification algorithms.

2. RELATED WORK

Computer networks are widely being used by industry, business and various fields of human life. Therefore, building reliable networks is a really important task for IT administrators. When considered the other way around, the rapid development of information technology produced several challenges to build reliable networks which are a really difficult task. There are several types of attacks threatening the availability, integrity and confidentiality of computer networks. Several research works are being carried out to get over this menace. Below are few, relevant to our work.

Ren *et al.* [5] put forward a Fuzzy C-Means clustering algorithm for intrusion detection. The algorithm was applied on six different subsets of KDD Cup 1999 data set with 5000 records each. The detection rate varies between 50.3% and 90.5% whereas the false positive rate ranges between 0.2% and 4.1%.

The work done by Mulay*et al.* [6] involves multiclass intrusion detection using support vector and decision tree. Their work was also evaluated on KDD Cup'99 data set.

Wang [10] also presented an improved K-Means algorithm to overcome the sensitivity problem of initial centre selection. The basic idea was to choose the initial centres as decentralized as possible. The improved algorithm was applied on the KDD Cup 1999 data set.

Clustering methods are commonly used for anomaly detection. Syarif et al. [11] proposed and discussed five different anomaly detection techniques. The authors used NSL-KDD data set for the evaluation of clustering algorithms in network anomaly detection

Govindarajan and Chandrasekaran [9] proposed an improved K Nearest Neighbor (KNN) algorithm for network anomaly detection application. Their proposed algorithm yields a reduction of the run time by up to 0.01 % and 0.06 % while error rates are lowered by up to 0.002 % and 0.03 % for normal and abnormal behavior respectively.
Wang et al. [12] proposed an intrusion detection framework

based on SVM and validated their method on the NSL–KDD dataset. They claimed that their method, which has a 99.92% effectiveness rate, was superior to the approaches; however, they did not mention used dataset statistics, number of training, and testing samples. Furthermore, the SVM performance decreases when large data are involved, and it is not an ideal choice for analyzing huge network traffic for intrusion detection.

The above work motivated us to work on 7 different machine learning algorithms to evaluate their metrics in order to find the best performing algorithm for the NSL-KDD Intrusion detection dataset.

3. EVALUATION METRICS

3.1 Accuracy

This metric is calculated by finding the total number of instances that are correctly predicted as positive cases to the total number of data that is present, the instances are classified into positive cases or negative cases by calculating the data that are divided into True positive (TP), True negative (TN), False positive (FP), False positive (FN).

$$\text{Accuracy} = TP+TN/TP+FP+FN+TN$$

3.2 Precision

It refers to the total data which is correctly predicted to be positive over the total number of data that are predicted to be positive, by observing the false positive and true positive instances, it can be calculated as:

$$\text{Precision} = TP/TP+FP$$

3.3 Recall (Sensitivity)

It is also known as the true positive rate (TPR), Sensitivity (SN) or detection rate. It indicates the total number of instances that are correctly predicted as positive over the total number of actually positive instances present. While detecting the overall positive data in the dataset the recall serves best as the main evaluation metric or the best performance indicator of positive data, it is calculated as follows:

$$\text{Recall} = TP/TP+FN$$

Precision and Recall are equally important for calculating the performance of the IDS, each individual is not enough to evaluate the performance of IDS.

4. DATASET USED (NSL-KDD)

For this project, we have worked on NSL-KDD dataset. KDDCUP1999 is the benchmark data set for intrusion detection. One of the most important deficiencies in the KDD data set is the huge number of redundant records, which lead to inaccuracy in the detection rate. There are two main reasons for the lack of efficiency in using KDDCUP99 data set. First one is the presence of lots of duplicates in training and testing records and the second one is the lack of difficulty measurement in records. Redundant records in training data set prevent learning methods from learning rare records such as U2R attack and R2L attack causing wrong results in testing data set which can wrongly increase the accuracy rate. With the simplicity of data set, detection methods can provide high accuracy without any trouble. [2]

NSL-KDD data set, developed by Tavallaee et al (2009), an enhanced version of KDDCUP1999 benchmark intrusion detection data set sought to solve the inherent problems of KDDCUP1999 data set. The first important limitation in the KDDCUP1999 data set is the huge number of redundant records in the sense that almost 78% training and 75% testing records are duplicated**.** Such limitations are being overcome by NSL-KDD.

NSL-KDD data set covers four major categories of attacks such as Probing attacks, Denial-of-Service (DoS) attacks, User-to-Root (U2R) attacks, and Remote-to-Local (R2L) attacks.

5. IMPLEMENTATION

Initially, import the data from the NSL-KDD dataset. Now perform all required pre-processing on the data to get it in a required format. After its extraction divide the available data into training and testing datasets.

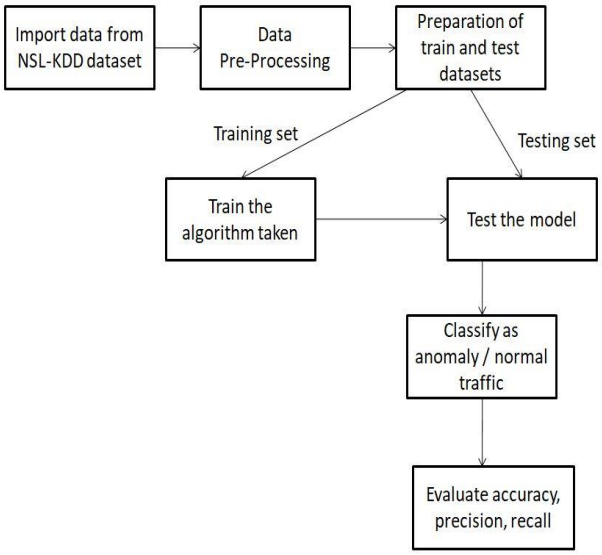

Fig. 1: System architecture

Train the algorithm taken with the training set. Now take a detection model, input the test data and the trained algorithmic input to it. This starts predicting the class to which the output belongs to i.e., either normal or anomalous. According to the output predicted and the actual output it has to predict, evaluate the metrics like accuracy, precision, recall. Continue the same process for all algorithms under consideration. Finally, rank them all based on their performance and find the best performing algorithm in a given context.

6. RESULTS

For each algorithm, we visualize the confusion matrix which shows the following widely used raw metrics.

- **TP (True Positive)**: this is defined as the number of anomalous traffic flows predicted is actually anomalous.
- **FP (False Positive)**: this is defined as the number of traffic flows predicted as anomalous but actually normal
- **TN (True Negative)**: this is defined as the number of traffic flows predicted as normal and actually normal
- **FN (False Negative)**: this is defined as the number of traffic flows predicted as anomalous but actually normal.

To conduct the experiments we have taken 20% of NSL-KDD dataset which has got 25,192 records of data with 38 features and 1 class column. This data is split randomly (for greater efficiency) into train and test data with a split ratio of 0.67. After splitting we have got 16,878 train records and 8,314 test records.

Table 1 shows the results when trained with entire train dataset and then evaluated with the test set. Accuracy is being calculated based on the number of right predictions. We also

generate a confusion matrix and a classification report which summarizes the number of true positives, false positives, true negatives, false negatives which are used in recall and precision calculation.

Table 1: Overall results

Algorithms	Accuracy	Precision	Recall
KNN	97.43	97	97
Naïve Bayes	73.15	92	74
Ensemble	89.23	86	81
SVM	85.57	84	81.5
RBF	68.93	69	66
K Means	67.67	68	35
Fuzzy C Means	74.69	72.1	38.6

Here among all given algorithms, KNN outperforms all other algorithms in terms of the evaluation metrics taken. The clustering algorithms are underperforming than the classification algorithms.

The results are being observed when graphs are plotted between the varying train sample size from figure 2 and accuracy, varying train sample size and training time from figure 3 for each algorithm. The train size is incremented 10% each time keeping the test set size constant i.e., 1000, 2000, 3000 4000,.....,16,000. It is observed that for all the algorithms with an increase in the train size the train time increased proportionally and the accuracy either increased or remained constant at a certain level with minimal fluctuations.

Fig. 2: Train sample size vs. accuracy

X-axis: No. of training samples taken
Y-axis: Accuracy

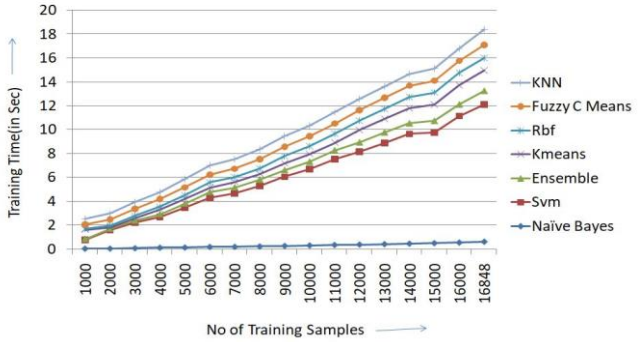

Fig. 3: Train sample size Vs training time

X-axis: No. of training samples taken
Y-axis: Training time (secs)

7. CONCLUSION AND FUTURE WORK

Intrusion detection and prevention are essential to current and future networks and information systems because our daily activities are heavily dependent on them. Furthermore, future challenges will become more daunting because of the Internet of Things. In this respect, intrusion detection systems have been important in the last few decades. Several techniques have been used in intrusion detection systems, but machine learning techniques are common in recent literature. Additionally, different machine learning techniques have been used, but some techniques are more suitable for analysing huge data for intrusion detection of the network. To address this problem, different machine learning techniques, namely, KMeans, K-Nearest Neighbors (KNN), Fuzzy C-Means (FCM), Support Vector Machine (SVM), Naïve-Bayes (NB), Radial Basis Function (RBF) and Ensemble method are investigated and compared in this work. KNN outperforms other approaches in accuracy, precision, and recall on the full train set samples from NSL-KDD dataset that comprise records being classified as normal and intrusive activities.

As a part of this project, we have taken anomaly-based intrusion detection systems under consideration for further observations because the previous works substantiate that anomaly-based systems perform better than the signature-based ones. This work can further be extended by using sensor flow and parallel processing platform i.e., GPU for more efficient and faster results.

8. ACKNOWLEDGEMENT

The authors would like to thank Mr Sajja. Rathan Kumar (Associate professor), Dept. of CSE for extending his support and guidance in the process of working on this paper.

9. REFERENCES

[1] L.Dhanabal, Dr S.P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms"

[2] "NSL-KDD data set analysis" by Anna University, Chennai

[3] Marzia Zaman, Chung Horng Lung "Evaluation of Machine Learning Techniques for Network Intrusion Detection"

[4] "Next Generation Intrusion Detection Systems", www.mcafeesecurity.com, Network Associates

[5] W. Ren, J. Cao, X. Wu, "Application of Network Intrusion Detection Based on Fuzzy C-Means Clustering Algorithm", Proc. of the 3rd International Symposium on Intelligent Information Technology Application, 2009

[6] S.A. Mulay, P. R. Devale, G.V. Garje, "Intrusion Detection System using Support Vector Machine and Decision Tree", International Journal of Computer Applications, vol. 3, no. 3, 2010.

[7] A. B-Perin, "Ensemble-based methods for intrusion detection", Master's thesis, Available online at http://code.ulb.ac.be/dbfiles/Bal2012 mastersthesis.pdf, Last accessed on August 7, 2017

[8] C-F Tsai, Y-F Hsu, C-Y Lin, W-Y Lin, "Intrusion detection by machine learning: A review", Journal on Expert Systems with Applications, vol. 36, 2009.

[9] M. Govindarajan and R. M. Chandrasekaran, "Intrusion detection using k-Nearest Neighbor", Proc. of the 1stInternational Conference on Advanced Computing, 2009

[10] S. Wang, "Research of Intrusion Detection Based on an Improved K-means Algorithm", Proc. of the 2nd International Conference on Innovations in Bio-inspired Computing and Applications, 2011

[11] Syarif I, Prugel Bennett A, Wills G., "Unsupervised clustering approach for network anomaly detection", Networked Digital Technologies Communications in Computer and Information Science, vol. 293. Berlin Heidelberg: Springer, 2012, pp.135–45

[12] H.Wang, J.Gu, and S.Wang, ''An effective intrusion detection framework based on SVM with feature augmentation,'' Knowl.-Based Syst., vol. 136, pp. 130–139, Nov. 2017, doi: 10.1016/j.knosys.2017.09.014

