Machine Learning Processing for Intrusion Detection

Ms. Vandana Vishnu Ardad, Ms. Ankita Dhar, Dr. Shikha Nema

[1]*Student, Dept. of Electronics Engineering, Usha Mittal Institute of Technology, Maharashtra, India*
[2]*Student, Dept. of Electronics Engineering, Usha Mittal Institute of Technology, Maharashtra, India*
[3]*Head of Department, Dept. of Electronics and communication, Usha Mittal Institute of Technology, Maharashtra, India*

---***---

Abstract - —*In current scenarios, data is one of the most valuable resource. The safety of data through any network is one of the most important factors for any system, company or organization. For better prediction of abnormal networks which can create troubles within the system or are threat to the valuable data, network intrusion detection systems are used. There are various machine learning techniques for many applications that works equally well on all data sets. In this paper five different Machine Learning algorithms are provided against KDD dataset with which intrusions can be detected. The accuracy of logistic regression, naive bays, support vector machine, K nearest neighbor and decision tree are discussed and analyzed with the help of ROC's. Among the various techniques decision tree gives optimum accuracy i.e. 99.83%.*

***Key Words*: IDS, Machine learning techniques, Confusion matrix, ROC's, Accuracy.**

1. INTRODUCTION

Data these days has developed to be the most important part of our day to day life. Be it our work, our bank transactions, our personal details or for business purposes. So its security is one of the major issues right now. The companies and business setups invest a lot and lot of money in cyber security or simply for the security of their data. Intrusion Detection system serve a pivotal role in securing computer networks.

IDS stand for Intrusion Detection System. To protect from cyber attacks many new advancements have been made like introduction of statistics or artificial intelligence.

Automatic intrusion detection system has become an important examination topic. Till now researchers have studied on Intrusion Detection Systems (IDS) successfully with the capability of detecting attacks in various fields; now latest on the scene are Machine Learning (ML) Approaches. In this paper a machine learning approach is used to distinguish between normal and abnormal network. ML processing algorithms can improve processing speed especially dealing with big data sets. Machine learning techniques are the sets of algorithms that have improved performance in the situations they have already encountered with vast range of applications such as speech recognition, pattern detection, outlier analysis etc. There are a number of machine learning techniques developed for many applications and there is no universal technique which work equally well on all data sets. In this work, we evaluated the machine learning algorithms for intrusion detection. The motive here is to evaluate classification accuracy of a supervised learning algorithm. In this paper the KDD data set is used for intrusion detection [1].

2. LITERATURE SURVEY

Traditional methods of IDS were based on low level attacks and generated isolated alerts. Although there was a logical connection between them but still it was incapable of giving combined results. To overcome this problem, DIDS was used [2]. The DIDS combines all these scattered alerts and makes use of their logical relationship, thus obtaining additional information. The huge range of small low cost embedded devices provided with one or more sensors are interconnected through wireless or cable network integrated to the internet. But the system used were not simple and used

Several technologies and network resources making it more complicated. Also due to this fact sophisticated management was necessary by learned and experienced administrators.

Further, sensor fusion was done to detect the abnormal Networks accurately [3]. IDS have distinct preferences for detecting a particular class of attack with improved accuracy while performing moderately on other classes. So deploying different methods on same network can give better and accurate detection. The main issue in this case was the false alarming. Handling of sensors was complex. Moreover the management was difficult as well. The process became slow as various sensors and methods were used.

To avoid the above complexities, a simple software application is used. Also programmable signature matching is used [4]. Networks use deep packet inspection to enable sophisticated services like intrusion detection. Signature matching is the heart of deep packet inspection which involves matching resupplied signature to network payloads at line rates. Content searching/ matching detects a variety of attacks. This method is efficient but does not use or apply machine learning capabilities to generalize signature matching techniques.

Further the network analysis here is done with the help of machine learning techniques. Various machine learning methods are applied on a data set known as KDD. The accuracy of different machine learning methods on this dataset is calculated. The machine learning approach is intelligent and involves various steps to distinguish between the normal and abnormal networks efficiently [1].

3. DATASET

The KDD dataset is a derived from big DARPA dataset used for intrusion detection.

3.1 Intrusion detector learning

A predictive model is built so that the data can be classified so as to detect an intrusion if present. The learning process includes building this model only. Its main agenda is to makes a model which is able to distinguish between good and bad network.
DARPA was prepared to evaluate the intrusion detection in 1998[5]. The KDD dataset is the version of this data set. Lincoln Labs worked for 9 weeks to acquire raw TCP dump data for a LAN. The LAN was tested on an Air Force Environment and preparation for many attacks was done. The training data was 4GB of compressed binary TCP data. This data was then processed for 5 million connection records. Each connection here is labeled as normal or attack type [6].

3.2 Attack types

1. DoS: Denial of service.
2. Probing: probing attack like port scanning
3. U2R: unauthorized access to the root.
4. R2L: unauthorized access from remote system.

syn flooding, port scanning, buffer overflow attacks, password guessing are the examples of DoS, Probbing, U2R and R2L respectively. This data set has 24 training attack and 14 types for the test data. Conclusion records used the destination host and the construction of window was made by 100 connections. In this model, 80% training set is used and 20% test set is there. Now for mining the unstructured data portions of packets automatically, few useful algorithms are there. Stolfo et al used domain knowledge to add features
which look for suspicious behavior in the data portions, like the number of failed login attempts. These features are known as”content” features. A few set of features are given in the three tables below which are defined for the connection records.

Table -1: Basic features of individual TCP connections

Feature Name	Description
duration	length of the connection
protocol_type	type of the protocol e.g tcp, ip etc.
service	destination network service, e.g., http, telnet, etc.
src_ bytes	from source to destination - number of data bytes
dst_bytes	from destination to source - number of data bytes
Flag	status of the connection = error or normal
land	if connection is from/to the same host = 1; 0 otherwise
wrong_ fragment	number of fragments which are ”wrong”
urgent	number of packets which are urgent

Table -2: Content features within a connection suggested by domain knowledge

Feature Name	Description
hot	number of indicators which are ”hot”
num_ failed	num failed logins failed login attempts
logged_ in	if successfully logged in = 1; 0 otherwise
num_ compromised	number of conditions which are ”compromised”
root_shell	if root shell is obtained = 1; 0 otherwise
su_ attempted	if ”su root” command attempted = 1 ; 0 otherwise
num_root	”root” accesses
num_file_ creations	number of operations for file creation
num_ shells	shell prompts
num_access_ files	operations on access control files
num _outbound_ cmds	in an ftp session, number of outbound commands.
is_ hot_login	if the login belongs to the ”hot” list = 1; 0 otherwise
is_ guest_ login	if the login is a ”guest” login = 1; 0 otherwise

Table -3: Traffic features computed using a two-second time Window

Feature Name Type	Description
count	in the past two seconds as the same connection as the current, the number of connections to the same host Note: The following features refer to these same host connections.
serror_ rate	percentage of connections that have "SYN" errors
rerror_ rate	percentage of connections with "REJ" errors
same_ srv_ rate	percentage of connections to the same service
diff_ srv_ rate	percentage of connections to different services
srv_ count	in the past two seconds as the same connection as the current, the number of connections to the same server Note: The following features refer to these same service connections.
srv_ serror_ rate	percentage of connections with "SYN" errors
srv_ rerror_ rate	percentage of connections with "REJ" errors
srv_ diff_ host_ rate	percentage of connections to different hosts

4. MACHINE LEARNING ALGORITHMS

4.1 Logistic Regression

It is used when only two outputs are there. It doesn't work on linear equation type of classes. For each experiment, only two values can be there. It produces a curve of outcomes having values between 0 and 1 only or yes/no [7].

The fundamental equation of linear model is:

$$g(E(y)) = \underline{\alpha} + ß\, x1 + x2 \quad (1)$$

The value lies between any ranges. But to compress the data to 0 and we used the sigmoid function for squashing.

Sigmoid Function of Logistic Function is:

$$g(x) = 1/(1 + e^{\wedge}(-x)) \quad (2)$$

4.2 Naive Bayes

This algorithm is based on the Bayes theorem used in probability. This classifier considers the features to be independent. The features independently contribute to the output probability.

This is one of the easiest to understand and fairly efficient way of classifying different type of classes as the calculation is easy. The classifier uses this simple formula based on Bayes theorem.

$$P(x/y) = p(y/x)p(x)/p(y) \quad (3)$$

Above,
P(x/y) is the later probability of the event.
P(x) is the former probability of the event.
P(y/x) is the likelihood
P(y) is the former probability of predictor.

4.3 Support Vector Machine

This classifier is based on separating two types of categories by using a hyperplane. Hyperplane is a feature by which two different classes which have different features are separated out. The hyperplane must have maximum distance between the data of both classes.[7]

The data, in the form of points, is plotted in a frame and best possible hyperplane is chosen to separate the two classes have distinct data points.

In SVM, the output taken from the linear function like linear regression is squashed and we get the values between -1 and 1.Here the main focus is on to maximize the distance between the hyperplane and data points. so to achieve this we use a cost function. The formula for cost function:

$$c(x, y, f(x)) = \{0,\ y * f(x) > 1\ 1\ else\ y * f(x), \quad (4)$$

4.4 K Nearest Neighbor

It is used for both linear and classification problems. In this method of classification, we choose a factor k i.e. the value of k is selected. And with the help of that value we create boundaries. These boundaries then help in distinguishing different kind of classes.[9] As the value of k is increased, the boundaries become smoother. Based on the majority votes of the neighbors, the outcome is processed. There is no assumption of the data so it is useful for the real time applications.

4.5 Decision Tree:

It is a graphical representation which is computed by the method of branching. The branches of various small possibilities are setup. Based on the homogeneity of the different sets a final set is created. It is a type of supervised learning. The samples are differentiated into two or more sets. The samples are segregated based on the values and then the best homogeneous set is created. This classifier is very easy to understand and it is one of the fastest ways to compute the outcome.[10]

5. IMPLEMENTATION

We used SPYDER software to run the machine learning algorithms. It writes and run codes in PYTHON language. For ease of understanding the codes for different classifiers are written in PYTHON. SPYDER has inbuilt functions and packages for classifiers. Hence this makes it easy to write and apply codes on this software. IT is a powerful interactive development environment for the Python language and also provides an object inspector that executes in the context of the console. Python includes numpy library which is used to acquire the numerical values. It also include package from library known as the pandas which is used to acquire the data set. With the help of this function we can access dataset for the code.[8] The train set and test set is divided and to avoid any randomness after that we put random state i.e 0. The average and mean values are pre calculated. Then classifiers use this data for their processing and next step includes importing of classifiers from the library. These classifiers then give the accuracy on data set. The predicted value is calculated using training data set and the test data set is used to make out the difference between actual and predicted values which gives us accuracy based on confusion matrix.

5.1 Confusion Matrix

A confusion matrix tells us about the overall efficiency and accuracy of an algorithm over a data set. It is based on test data sets for which the true values are known. It is an efficient way of calculating the accuracy. It gives us the knowledge about the true positive and true negative values and also false positive and false negative values of a particular classifier over a data set. Therefore confusion matrix is as follows:

$$cm = [TP\ FP\ FN\ TN\ ]$$

Where
TP : True Positive data
TN : True Negative data
FP : False Positive data
FN : False Negative data

5.2 Accuracy is given as follows

Accuracy=(TruePositive+TrueNegative)/(TotalData Samples) (5)

5.3 Receiver Operating Characteristic curve (ROC curve)

The Receiver Operating Characteristic curve is obtained by plotting the rate of true positive value (TPR) versus the rate of false positive value (FPR) at different threshold settings. In machine learning the true-positive rate is also known as sensitivity, or probability of detection and the false-positive rate is known as the fall out or probability of false alarm which is calculated as 1 specifically. In general, the ROC curve can be generated if the probability distributions of detection as well as false alarm are known, by plotting the cumulative distribution function of the detection (probability values) in the y axis against the cumulative distribution function of the false alarm (probability values) on the x axis.

6. RESULT

6.1 Confusion Matrix

Table-4: Confusion Matrix table

Classifier Type	Confusion Matrix
Logistic Regression	$[10980\ 758\ 472\ 12985\ ]$
Naïve Bayes	$[10831\ 907\ 1594\ 11863\ ]$
Support Vector Machine	$[11033\ 705\ 431\ 13026\ ]$
KNN	$[11660\ 78\ 45\ 13412\ ]$
Decision Tree	$[11718\ 20\ 22\ 13435\ ]$

6.2 Accuracy

Table-5: Accuracy Table:

Classifier Used	Accuracy
Logistic Regression	95%
Naïve Bayes	90%
Support Vector Machine	95.5%
KNN	99.51%
Decision Tree	99.83%

6.3 Receiver Operating Characteristic Curves

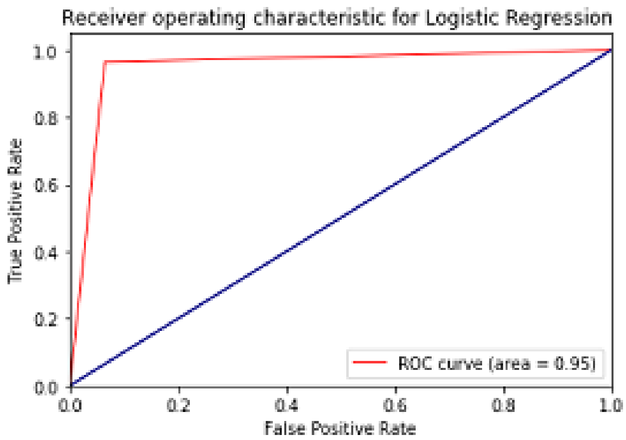

Fig -1: ROC of Logistic Regression

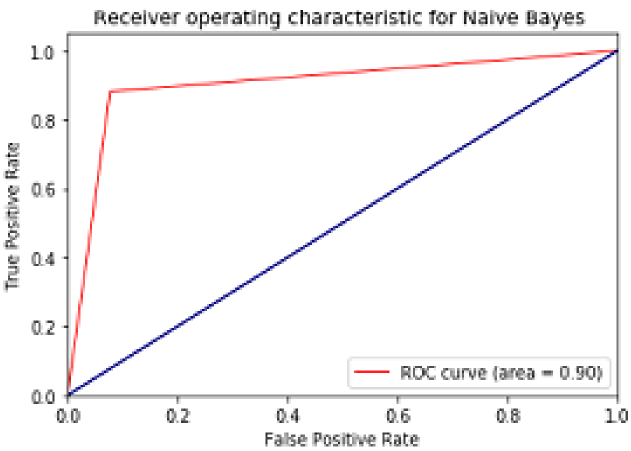

Fig -2: ROC of Naïve Bayes

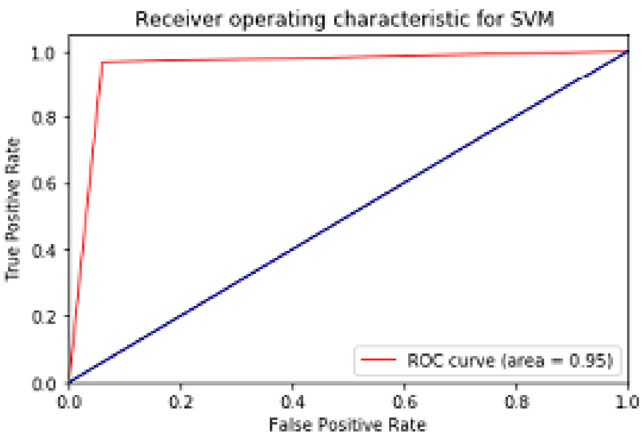

Fig -3: ROC of Support Vector Machine

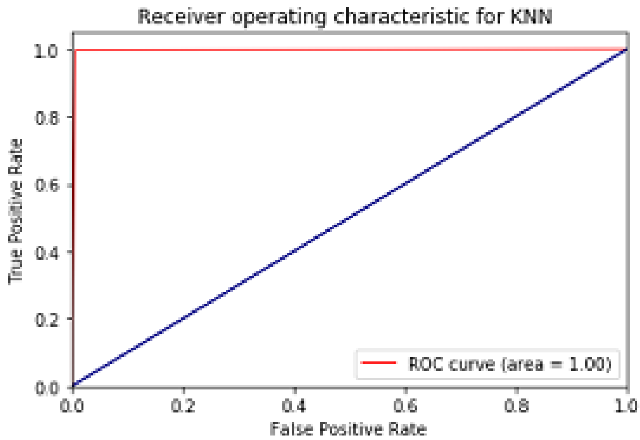

Fig -4: ROC of KNN

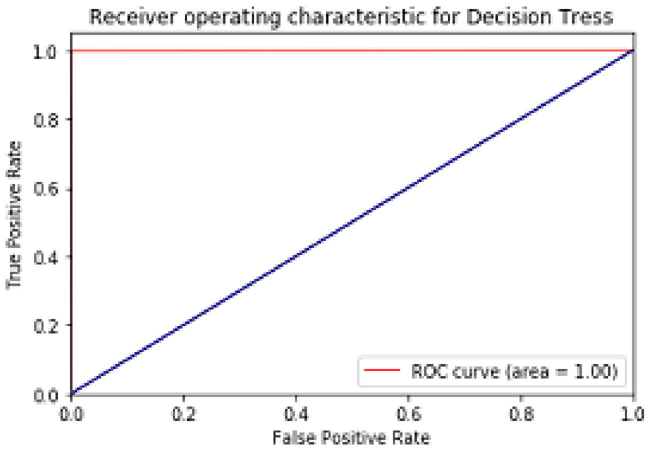

Fig -5: ROC of Decision Tree

7. CONCLUSION

The machine learning techniques which were applied on KDD dataset is studied. Logistic regression, naive bayes, support vector machine, K nearest neighbor and decision tree classifiers are used and their result is calculated. We obtained confusion matrices for all classifiers we used and calculated the Accuracy of each classifier. The ROCs are also developed to better understand the machine learning efficiency on given data set.

Among the various techniques, Decision Tree classifier has the optimum accuracy i.e 99.83% as it is best suited for outputs with two values. The branching techniques are best suited for the classification of networks. This can be further used in user interfaces. In which the user inputs the name of network through voice or text and in the background the machine learning algorithms detect the the network to be normal or abnormal. The output is displayed to the user through text or through voice. The real time use is yet not possible. But the study of machine learning algorithms can be done through this process. The machine learning techniques thus can be used to detect intrusions and they have a good efficiency and accuracy as well.

REFERENCES

[1] Machine Learning techniques for Intrusion detection: A comparative Analysis. Yasir Hamid, M Sugumaran, Ludovic Journaux. November 2016.
[2] Network intrusion Traditional detection system embedded on smart sensors. IEEE Trans. on Industrial electronics, March 2011.
[3] Improvement in intrusion detection with advance in sensor fusion, IEEE Trans. on information forensics and security, September 2009
[4] Evaluating GPU for network packet signature matching, in March, 2009.
[5] Usefulness of DARPA dataset for Intrusion Detection System.
[6] Cost- based modeling and evaluation for data mining with application to fraud and Intrusion Detection.
[7] Supervised Machine Learning: A Review of Classification Techniques by S. B. Kotsiantis.
[8] R. Heady, G. F. Luger, A. Maccabe, and M. Servilla, The architecture of a network level intrusion detection system. Department of Computer Science, College of Engineering, University of New Mexico, 1990.
[9] W. Lee, S. J. Stolfo, and others, Data mining approaches for intrusion detection, in Usenix security, 1998.
[10] M. Khan and S. M. K. Quadri, Evaluating Various Learning Techniques for Efficiency, International Journal of Engineering and Advanced Technology (IJEAT), vol. 2, no. 2, pp. 326331, 2012.