Network Intrusion Detection using Machine Learning.

A PROJECT REPORT

Submitted by

Anshuman Dey Kirty (15BCE0408)

CSE4020

Machine Leaning

In partial fulfilment for the award of the degree of

B.Tech

in

Computer Science and Engineering

Under the guidance of

Vijayasherly V

Assistant Professor, SCOPE,

VIT University, Vellore.

VIT® UNIVERSITY
(Estd. u/s 3 of UGC Act 1956)
VELLORE ▪ CHENNAI
www.vit.ac.in

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

March, 2018

ABSTRACT

The need to secure networks has increased as the number of people connecting to the network are increasing rapidly and using networks for storing or accessing critical information. In this paper we have assessed and compared various machine learning algorithm and then propose a system based on the best performing algorithm. Our system is an intrusion prediction system with low error rate and can be implemented in real world. The dataset used in this project would be the database that contains set of data to be assessed, which includes a wide variety of intrusions simulated in a military network environment. The dataset contains mainly the normal state, DDoS and some other attacks. The system will not only predict a malicious network but also point to exactly under which type of attack the network is subjected to.

Key words: learning, decision table, random forest, network intrusion.

INDEX

Topic	Page No.
1. Introduction	5
2. Background of the Work	5
3. Overview of the Work	7
3.1.Problem description	7
3.2.Dataset Description	7
3.3.Working model	8
4. Implementation	10
4.1.Modules/Programs	10
4.2.Source Code	10
4.3.Execution snapshots	11
5. Conclusion and Future Directions	12
6. References	14

List of Tables

Table 1 KDD Dataset Attacks..8

Table 2 R Output...11

Table 3 Predicted Output for Test Data 11

Table 4 Random Forest Output Graph..11

Table 5 Weka Result...12

Table 6 Algorithm Evaluation Graph 12

Table 7 Algorithm Evaluation Graph 13

1. Introduction

With the advancement in the technology, millions of people are now connected with each other through one or other form of network where they share lots of important data. Hence the need of security to safeguard data integrity and confidentiality is increased rapidly. Although effort have been made to secure data transmission but at the same time, attack technique for breaching the network continued to evolve. Thus it leads to the need of such a system which can adapt with this ever changing attack techniques. In this paper, we have purposed a system which is based on machine learning. Our aim is to find the based suitable machine learning algorithm which can predict the type of network attack with highest accuracy and then develop a system which uses this algorithm to detect network intrusion. The algorithms which we have compared are Naïve Bayes, Decision Table, K Nearest Neighbor, Random Forest and Adaboost. The dataset used for training the model is KDD 99 dataset. The reason why we have used machine learning is the flexibility that it provides to the system for example, if any new type of attack is developed in future the system can be trained for predicting that attack. There are a few types of intrusion detection system out of which ours is a knowledge based intrusion detection system which is also known as the anomaly based system. It registers the anomalies and in future predicts such malicious network to send out an alert. This way the network can disconnect to the such a connection and then have only secured connections.

2. Literature Survey

In [1], Rafath and D Vasumathi, classified the intrusion detection system into two types namely Network based IDS and Host IDS. The latter monitors all the activities of inspected packets and resources that are being utilized by the programs. In case of any alteration in networks, the user gets a network alert. HIDS is incorporated into the computer framework to detect the abnormalities and protect the information from the intruder. On the other hand, NIDS is the attribute function of target system. It uses anti-thread software to control incoming and outgoing threads. It consists of signature-based classification, which help in identifying the abnormalities by comparing it with log files and previous signature.

In [2], the authors proposed an AI based Intrusion detection system using a deep neural network. Neural networks consisting of four hidden layers and 100 hidden units was used for the intrusion detection system. They used non-linear ReLU as the activation function for the hidden layer neurons to enhance the model's performance. They adopt stochastic optimization method for learning in DNN. For the training and testing of their model they used KDD CUP 99 dataset. They were able to reach the accuracy of 99% for all the cases.

In [3], they have proposed a NIDS (Network Intrusion Detection System) which is based on a feature selection method called Recursive Feature Addition (RFA) and bigram technique. They tested the model on the ISCX 2012 data set. Moreover, they have proposed a bigram technique to encode payload string features into a useful representation that can be used in feature selection. They have also proposed a new evaluation metric called that combines accuracy, detection rate and false alarm rate in a way that helps in comparing different systems and selecting the best among them.

In [4], they have proposed a new intrusion detection system and addressed the problem of adaptability in the field of intrusion detection. The proposed IDS is an adaptive solution which provides the capability of detecting known and novel attacks as well as being updated according to the new input from human experts in a cost-effective manner.

[5], it deals with the evaluation and statistical analysis of labelled flow based CIDDS-001 dataset used for evaluating Anomaly based (NIDS) Network Intrusion Detection Systems. They basically used two techniques, k-means clustering and k-nearest neighbor classification to measure the complexity in terms of prominent metrics. Based on evaluation, they concluded that both k-means clustering k-nearest neighbor classification perform well over CIDDS-001 dataset in terms of used prominent metrics. Hence the dataset can be used for the evaluation of Anomaly based Network Intrusion Detection Systems.

In [6], The IDS is based on anomaly detection method. In such technique, a system tries to estimate the ‘normal’ state of the network and generates an alert when any activities deviate from this ‘normal’ state. The main benefit of anomaly-based system is that it is able to detect previously unseen intrusion events. They have classified detection techniques into three categories statistical based, knowledge-based, and machine learning-based. In statistical based technique, a random viewpoint is used to represent the behavior of the system. While knowledge based technique, utilize the available system data to capture the behavior of system. Finally, the machine learning based technique uses an explicit or implicit model to enable categorization of the analyzed pattern.

In [7], various machine-learning techniques can result in higher detection rates, lower false alarm rates, reasonable computation, and communication costs in intrusion detection. In this paper, Mahdi Zamani and Mahnush Movahedi studied several such technique and schemes to compare all their performance. They divide the schemes into methods based on classical computational intelligence (CI) and artificial intelligence (AI). They explain how several features of CI techniques can be used to build modern and efficient IDS.

In [8], firstly, network attacks are identified and the performance of the algorithms are compared. The Dimension Reduction focuses on using information obtained KDD Cup 99 data set for the selection of attributes to identify the type of attacks. The dimensionality reduction is firstly performed on 41 attributes to 14 and 7 attributes based on Best First Search method and then two-classification algorithm are applied.

In [9], Mohammad Saiful Islam Mamun and A.F.M. Sultanul Kabir proposed a hierarchical architectural design based intrusion detection system that fits the restrictions and present demands of wireless ad hoc sensor network. In their proposed intrusion detection system architecture, they followed clustering mechanism to build a four level hierarchical network that increases network scalability to large geographical area and use both anomaly detection and misuse techniques for intrusion detection. They introduced intrusion response together with GSM cell concept as well as policy based detection mechanism for intrusion detection architecture.

3. Overview of the Work

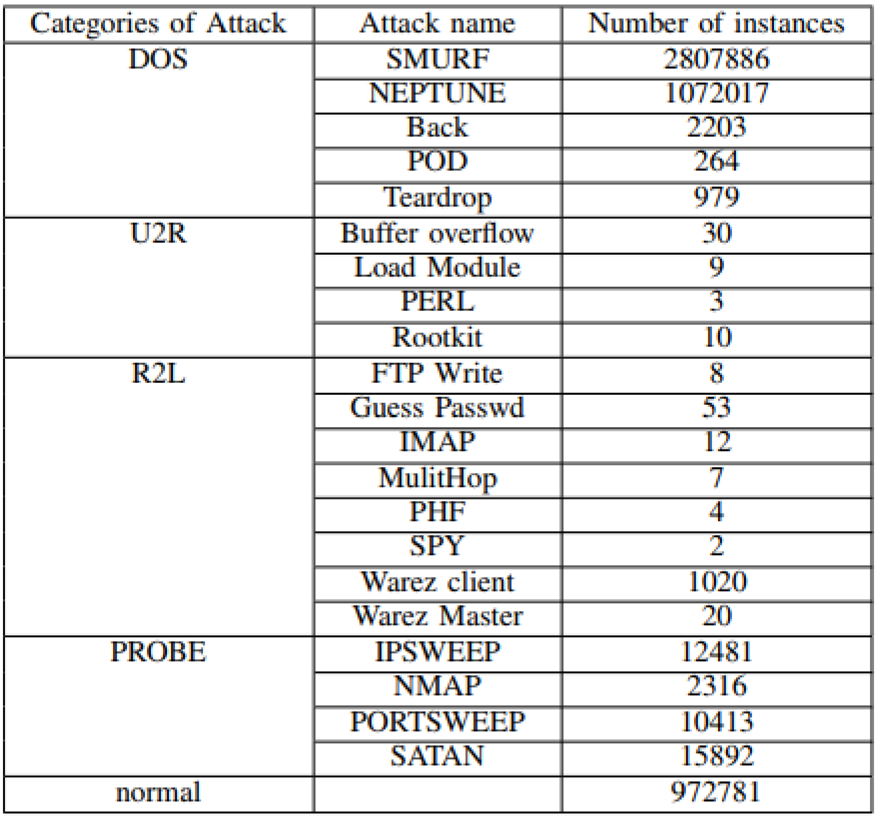
3.1 Problem Description

With the rapid development of information technology in the past two decades. Computer networks are widely used by industry, business and various fields of the human life. Therefore, building reliable networks is a very important task for IT administrators. On the other hand, the rapid development of information technology produced several challenges to build reliable networks which is a very difficult task. There are many types of attacks threatening the availability, integrity and confidentiality of computer networks. The Denial of service attack (DOS) considered as one of the most common harmful attacks.

3.2 Dataset Description

The dataset being used is the KDD99 dataset for network intrusion. It’s a famous dataset being used by many researchers for the purpose of intrusion detection applying various learnings. The dataset contains many attack types like the DOS, U2R, R2L, Probe and normal (no attack). There are 21 type of attacks inside the main categories mentioned above.

Table 1 KDD Dataset Attacks



Categories of Attack	Attack name	Number of instances
DOS	SMURF	2807886
	NEPTUNE	1072017
	Back	2203
	POD	264
	Teardrop	979
U2R	Buffer overflow	30
	Load Module	9
	PERL	3
	Rootkit	10
R2L	FTP Write	8
	Guess Passwd	53
	IMAP	12
	MulitHop	7
	PHF	4
	SPY	2
	Warez client	1020
	Warez Master	20
PROBE	IPSWEEP	12481
	NMAP	2316
	PORTSWEEP	10413
	SATAN	15892
normal		972781

The dataset contains a total of 41 attributes which could be used to determine if the attack is malicious or not at all an attack.

3.3 Working Model

The dataset taken from the Kdd99 is a huge dataset and the one that we have used in our research is under the folder corrected. Our aim is to not only to find the best algorithm suited for the intrusion detection but also to implement it using the programming language R. The first process of applying learning is to pre-process the data. First, we convert the files to CSV format. Then we have to remove the redundant rows from the dataset. Then our next step is to see whether there are any missing values and then to remove those corresponding rows too. The next process is to use this dataset and put it across various machine-learning algorithms that might give good results by correctly classifying the instances. The tool that we used is the Weka. Weka is an open source Java platform for processing, classifying, clustering and visualization. It is considered as one of the better data mining tools and therefore we have used it. Steps involved in using Weka are

1. Importing the dataset
2. Classifying and choosing the algorithm.
3. Using the *10 Fold* method
4. Again testing using the *Percentage Split* (70%)
5. Checking the *Correctly Classified Instance* Percentage.

The values are observed and tabulated shown in the Results below.

It can clearly be seen from the above give values that the best algorithm that can be used for the network intrusion detection is the Random Forest. Random Forest algorithm is a classification algorithm based on ensemble learning. It works by building multiple decision trees at training and the developed decision trees forms the output function.

The algorithm is elected and now the implementation using the R programming language is to be done. The drawback of this intensive and the accurate algorithm in this case is that the computation time is very high. To lessen the computational time, we use feature selection algorithm. The feature selection algorithm used is the InfoGainAttribute using the Weka tool. Information gain is a feature selection method uses entropy of the class variable and then assess the feature.

Using this method, we expect no considerable drop in accuracy in terms of the percentage of correctly classified classes and a great reduction in time taken to detect intrusion in the network. This makes the system for intrusion detection more efficient.

The attributes elected are used in the R program to predict the error rate and in future to predict if a network is bad or normal. This program when developed fully will act as a filter to determine if a network is secure and will continuously learn from its own series of data making it better and stronger with each type of attack.

4. Implementation

4.1 Modules

The project mainly consists of 2 main modules:

i) The Algorithm testing using WEKA

ii) Feature/Attribute selection using the *InfoGain* Algorithm

iii) The implementation on the Algorithm using R.

4.2 Source Code

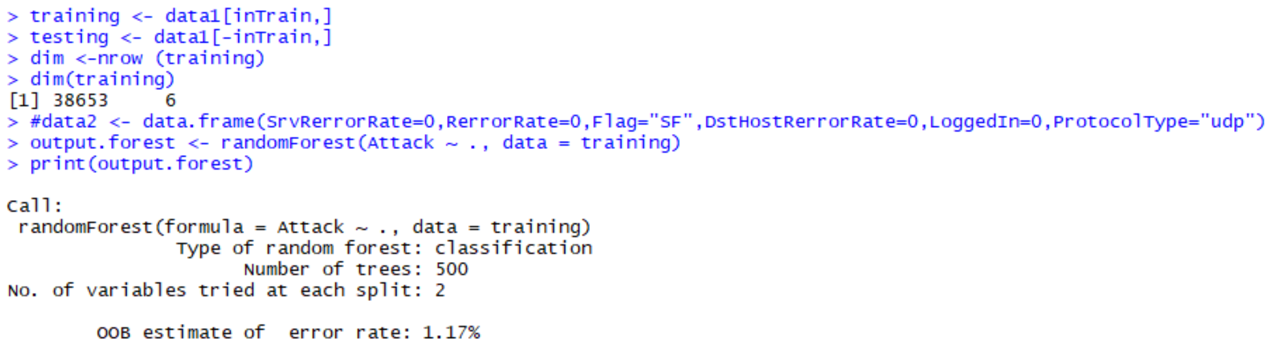
R code

```
library(randomForest)
#install.packages("caret")
library(caret)
library(e1071)
data <- read.csv
(file="C:\\Users\\ANSHUMAN\\Desktop\\Semester 6\\Cyber
Security\\Project\\corrected\\Book2.csv",header=T)
data1 <- data[,c("SrcBytes", "DstBytes",
"DstHostSameSrvRate", "Count",
"DstHostDiffSrvRate","Attack" )]
inTrain <- createDataPartition(y=data1$Attack,p=0.5,
list=FALSE)
str (data)
training <- data1[inTrain,]
testing <- data1[-inTrain,]
dim <-nrow (training)
dim(training)
#data2 <-
data.frame(SrvRerrorRate=0,RerrorRate=0,Flag="SF",DstHost
RerrorRate=0,LoggedIn=0,ProtocolType="udp")
output.forest <- randomForest(Attack ~ ., data =
training)
print(output.forest)
plot (output.forest)
```

```
pred <- predict(output.forest,testing)
pred
str (data)
```

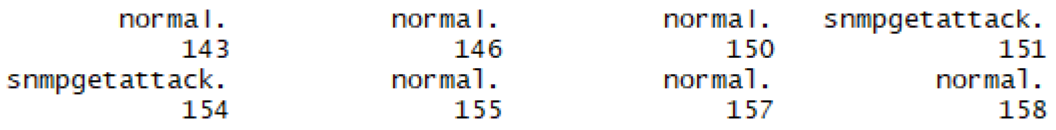
4.3 Execution Snapshot

Table 2 R Output



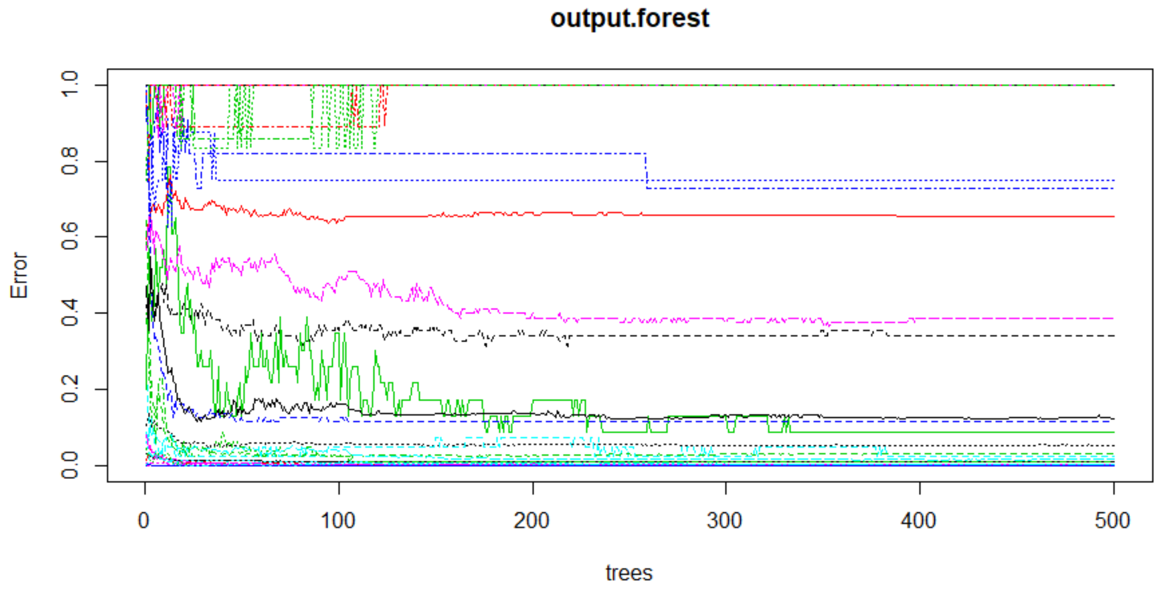
```
> training <- data1[inTrain,]
> testing <- data1[-inTrain,]
> dim <-nrow (training)
> dim(training)
[1] 38653     6
> #data2 <- data.frame(SrvRerrorRate=0,RerrorRate=0,Flag="SF",DstHostRerrorRate=0,LoggedIn=0,ProtocolType="udp")
> output.forest <- randomForest(Attack ~ ., data = training)
> print(output.forest)

Call:
 randomForest(formula = Attack ~ ., data = training)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 1.17%
```

Table 3 Predicted Output for Test Data



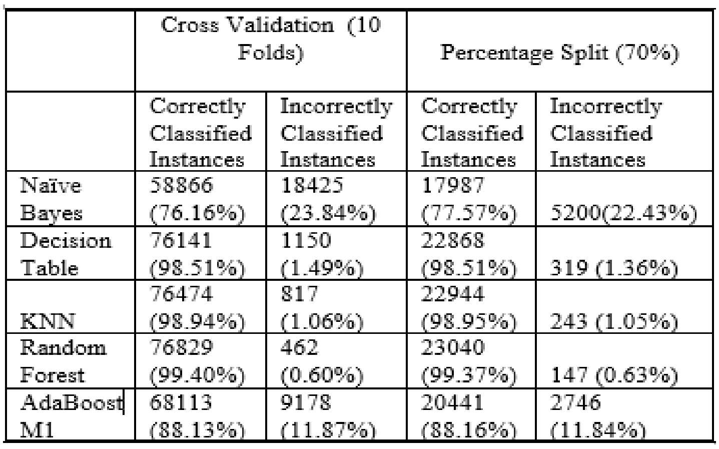
```
          normal.            normal.            normal.   snmpgetattack.
              143                146                150              151
   snmpgetattack.            normal.            normal.          normal.
              154                155                157              158
```

Table 4 Random Forest Output Graph



5. Conclusion

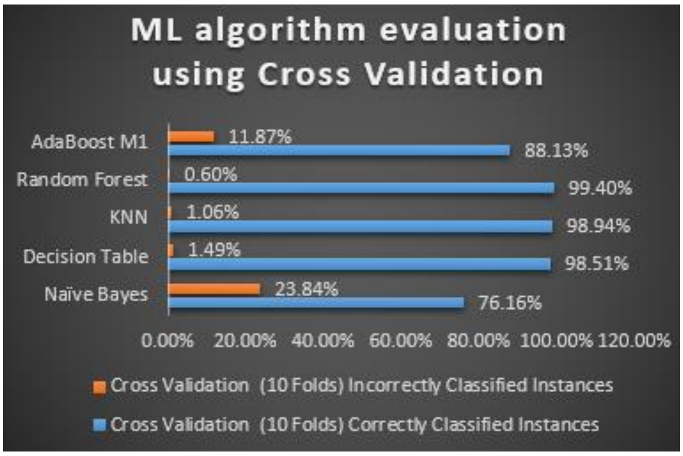
Table 5 Weka Result



	Cross Validation (10 Folds)		Percentage Split (70%)	
	Correctly Classified Instances	Incorrectly Classified Instances	Correctly Classified Instances	Incorrectly Classified Instances
Naïve Bayes	58866 (76.16%)	18425 (23.84%)	17987 (77.57%)	5200(22.43%)
Decision Table	76141 (98.51%)	1150 (1.49%)	22868 (98.51%)	319 (1.36%)
KNN	76474 (98.94%)	817 (1.06%)	22944 (98.95%)	243 (1.05%)
Random Forest	76829 (99.40%)	462 (0.60%)	23040 (99.37%)	147 (0.63%)
AdaBoost M1	68113 (88.13%)	9178 (11.87%)	20441 (88.16%)	2746 (11.84%)

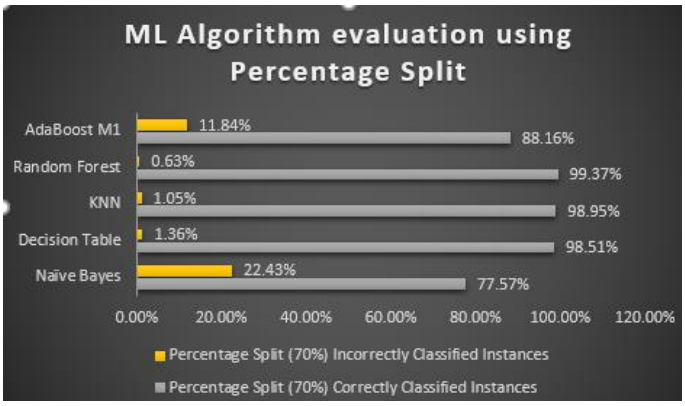
The Random Forest, KNN and the Decision Table give high accuracy but the accuracy achieved in case of the Random Forest is the highest. Therefore, to further our project we choose the Random Forest algorithm, use it practically to find out the error rate using the R Programming language.

Table 6 Algorithm Evaluation Graph



The graph in Figure 1 shows that the accuracy for the Random Forest algorithm is 99.40%, the highest. It has been evaluated using the 10 Fold Cross Validation method. In this method the dataset for training is divided into 10 parts and for each part the other 9 parts acts as the training set and the 1 part as the testing set. Average of all such sets gives the 10 Fold validation.

Table 7 Algorithm Evaluation Graph



The graph in Figure 2 shows that the accuracy for the Random Forest algorithm is 99.37%, the highest. It has been evaluated using the Percentage split (70) method. In this method the dataset for training is divided 70 percent for training purpose and the other 30 percent for the testing purpose.

Before directly using all the 41 attributes onto the R we do feature selection to select the attributes using the InfoGain method to reduce the time of computation whereas the accuracy is not much compromised. The attributes elected are

- SrcBytes - number of data bytes from source to destination
- DstBytes - number of data bytes from destination to source
- DstHostSameSrvRate – Destination host same server rate.
- Count - number of connections to the same host as the current connection in the past two seconds
- DstHostDiffSrvRate - Destination host different server rate.

The number of attributes reduce from 41 to 5 using the InfoGain method which makes it much faster in terms of execution time.

The R language is then used to see the OOB Estimate of Error that is retrieved after using the Random Forest machine-learning algorithm. The OOB Estimate of Error was found to be 1.46% when the half of dataset was used in training and the other half in testing the algorithm. For each of the testing dataset we can predict if the connection setup is normal or malicious and even pin point to the particular type of attack (example – Neptune, smurf, Saturn, teardrop, rootkit etc.) This means that we can be 98.54% sure of the prediction value shown by the algorithm running on R.

6. References

[1] Review on Anomaly based Network Intrusion Detection System Rafath Samrin Computer Science and Engineering ISL Engineering College Hyderabad, India D Vasumathi Computer Science and Engineering JNTUH Hyderabad, India.

[2] Method of Intrusion Detection using Deep Neural Network by Jin Kim, Nara Shin, Seung Yeon Jo and Sang Hyun Kim.

[3] Network intrusion detection system based on recursive feature addition and bigram technique.

[4] Adaptive and online network intrusion detection system using clustering and Extreme Learning Machines.

[5] Statistical analysis of CIDDS-001 dataset for Network Intrusion Detection Systems using Distance-based Machine Learning.

[6]Anomaly-based network intrusion detection: Techniques, systems and challenges by P. Garcı´a-Teodoroa, J. Dı´az-Verdejoa, G. Macia´-Ferna´ndeza, E. Va´zquezb.

[7] Machine Learning Techniques for Intrusion Detection Mahdi Zamani and Mahnush Movahedi.

[8] Network Intrusion Detection System Using Reduced Dimensionality

[9] Hierarchical Design Based Intrusion Detection System For Wireless Ad Hoc Sensor Network.