Stream Data Mining for Self Learning Networks

Sunil Angadi (M.Tech)
Department of Computer Science and Engineering,
Amrita School of Engineering, Bangalore
sunilangadi2@gmail.com

Mr.G Radhakrishnan
Assistant Professor, CSE Dept
Amrita School of Engineering, Bangalore
g_radhakrishnan@blr.amrita.edu

***Abstract*— In this paper it classifies network traffic data as binary classification of Benign versus Malicious. for the original NSL-KDD dataset using Artificial Neural Network, The effectiveness of the proposed algorithm is compared with Tree Based Classifier, logistic regression, support vector machine and the validity of the proposed algorithm is verified by using MATLAB. Moreover, a taxonomy and survey of artificial neural network classification system is presented based on current and previous work.**

***Keywords*— Artificial Neural Networks, Deep Packet Inspection, Self Learning Networks, Line search, Trust Region**

I. INTRODUCTION

The network traffic is heterogeneous and it has traffic of variety of applications and utilities. These applications are different and it needs there on specification with respect to network parameters (e.g. Bandwidth, jitter, etc). The effectiveness of these applications will be reduced if these network parameters are not satisfied, making these requirements in a normal Local Area Network (LAN) with its good amount of bandwidth might be easy, but when considering internet connection with WANs its usually a challenging task because of network traffic and bandwidth constraints.

Network traffic classification helps to identify various applications along with protocols that exist in a network. Different methods such as network control, discovery, monitoring, and optimization can be done on the identified traffic in order to enhance the network performance and utilization. Typically, once the packets get classified or identified as belongs to a particular application or protocol, these classified packets are flagged or marked. These markings on the particular packet help the router to determine service policies to be applied for the network routing process. All generic classification methods based on source IP address, destination IP address and IP protocol, etc. these are drawback in their ability as the inspection is limited to the IP header only. Similarly, classification based on transport layer ports also limited [1]. The problem with port based classification approach is that all current applications not using standard ports or registered ports. Some applications themselves use well defined ports of other applications. Hence the transport layer based port mechanism of application identification and classification is not always reliable and feasible.

A) Deep Packet Inspection

Deep Packet Inspection (DPI) is a form of computer network packet filtering technique that examines the header information and data part of a packet it enters to an inspection point, finding for protocol noncompliance, spam, viruses, intrusions, or defined criteria to decide whether it has to transfer to next router or not and it collects some information that functions at the Application layer of the OSI (Open Systems Interconnection) model. Deep Packet Inspection provides advanced network management, security functions, user management as well as internet data mining, internet censorship and eavesdropping [5]. Although DPI has been used for Internet management for many years, some advocates of net neutrality fear that the technique may be used to reduce the openness of the Internet. Network management is a part of network traffic and security engineering. Deep Packet Inspection depends on the availability of training dataset and it needs a cumbersome overhead of updating dataset to address these issues our proposed system uses Artificial Neural Network which learns the features automatically without the need of regular updating dataset.

B) Artificial Neural Networks (ANN)

An ANN uses a collection of connected units called artificial neurons these algorithms are inspired by biological neurons in a human brain. Each synapse between neurons can transfer a signal to another neuron. The receiving neuron called postsynaptic it can process the signals to the downstream neurons connected to it. In common ANN implementation, the synapse signal is a real number, and the result of each neuron is evaluated by a non-linear function of the sum of all its input. Synapses and neurons may also have a weight that varies as learning proceeds, which can decrease or increase the strength of the signal that it sends to next. Further, they will have a threshold such that only if the aggregate signal is below or above that level is the downstream or upstream signals sent respectively, typically neurons are arranged in layers. Each layer may perform different methods of transformations on their inputs. And Signals travel from the input layer to output layer, possibly after traversing the layers at multiple times.

II. RELATED WORK

Many of the researchers are working on the application of Machine Learning (ML) techniques with Artificial Intelligence discipline for internet traffic classification. The machine learning technique involves a series of steps. Major step is features are defined in such a way that future unknown internet traffic with malicious can be predicted and differentiated. Features are attributes of traffic flows computed over series of packets such as minimum or maximum packet length, flow duration and packet inter-arrival times, payload length etc, using this stream dataset by creating some association rules along with good learning rate way train the designed machine learning model to predict this particular internet traffic as malicious. Each ML algorithm has a unique approach for selecting and assigning a priority set of features, which eventually gives different dynamic results during training and testing. Self Learning Networks (SLN) architecture is an advance technology that combines powerful data analytics and wide set of machine learning technologies along with advanced networking, to achieve the network to become predictive, adaptive, proactive, and overall results into intelligent network. SLN architecture relies on the use of distributed, lightweight, yet complex analytic engines, referred to as Distributed Learning Agents (DLAs), implemented across the network. The DLA reports findings to the SLN Orchestrator, which provides orchestration and interaction among the DLAs, as well as supports additional features to be developed in the future.

A. Port based IP traffic classification

TCP and UDP gives flexible approach to multiplexing of multiple network flows among common IP endpoints through the use of fixed port numbers. Various software applications use 'well known' port on the local host. In this port based classifier uses TCP three-way handshake for session establishment and to know the server side of a client-server TCP connection establishment. Once connection is established The application is finally inferred by looking up the TCP SYN packet with target registered port number in the Internet Assigned Numbers Authority (IANA) list of registered ports. UDP also uses ports similarly but though without connection establishment not either the maintenance of connection state the major drawback of this approach is new applications and protocol don't follow these standard registered ports so in the prediction error rate is high[4].

B. Payload based IP traffic classification

To enhance the reliability on the semantics of registered port numbers, nowadays many industry products utilise state full reconstruction of session and application information from each packet's content. In this Payload based traffic classification of P2P (Peer to Peer) traffic by checking the signatures of the packet at the application level and it could reduce false positives and false negatives to 5% of total bytes for most P2P protocols so far studied. The traffic flows port number is verified when traffic classification procedure begins. If no well-known registered port is used, and the flow is passed to next stage in the second stage, the first packet is verified to see whether it contains a valid known signature. If it is not found, then the packet is examined to see whether it has a well-known protocol. If these tests cases fail, then the protocol signatures in the first Kbytes of the network flow are checked. Traffic flow remains unclassified after this stage it will require inspection of the entire flow payload. In this classification result shows that payload based classification is capable of correctly classifying 69% of the total bytes [4].In this today's world of machine learning prediction models having accuracy of this much less is not acceptable. So in order to overcome this we are using ANN based classification.

III. CLASSIFICATION USING ANN

Dataset is collected from the communication network and it is a sequence of TCP packets with various network traffic parameters between these interval data flows from a source IP address to destination IP address under some well defined protocol with some specific network constraints. Each connection is named as either attack or normal in the output class label for each connection record it consists of information around 100 bytes.

NSL-KDD dataset is used to detect attacks on four attack categories such as Denial of Service [Dos], Probing, Remote to Local [R2L], and User to Root [U2R].

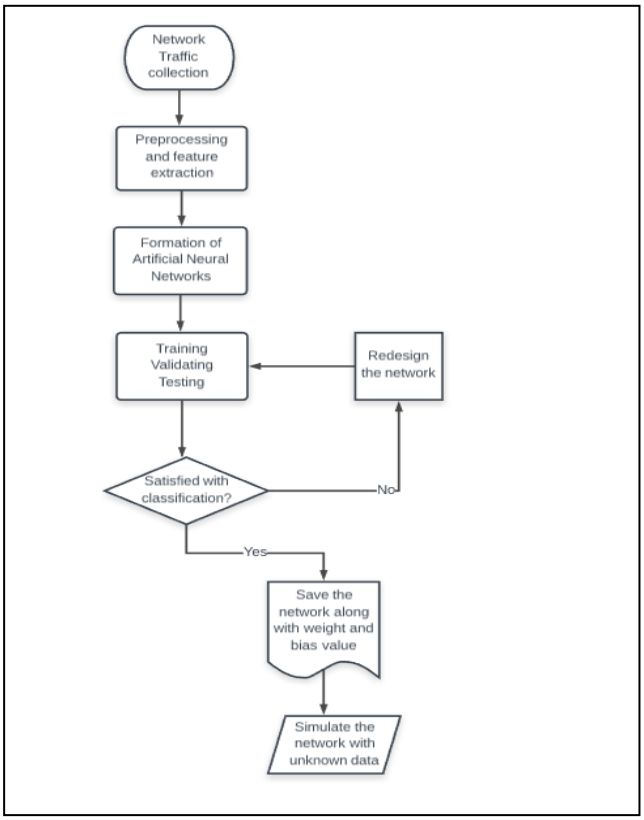

Figure 1.Model of ANN classification process

Type	Features
Nominal	Protocol_type(2), Service(3), Flag(4)
Binary	Land(7), logged_in(12), root_shell(14), su_attempted(15), is_host_login(21),, is_guest_login(22)
Numeric	Duration(1), src_bytes(5), dst_bytes(6), wrong_fragment(8), urgent(9), hot(10), num_failed_logins(11), num_compromised(13), num_root(16), num_file_creations(17), num_shells(18), num_access_files(19), num_outbound_cmds(20), count(23) srv_count(24), serror_rate(25), srv_serror_rate(26), rerror_rate(27), srv_rerror_rate(28), same_srv_rate(29) diff_srv_rate(30), srv_diff_host_rate(31), dst_host_count(32), dst_host_srv_count(33), dst_host_same_srv_rate(34), dst_host_diff_srv_rate(35), dst_host_same_src_port_rate(36), dst_host_srv_diff_host_rate(37), dst_host_serror_rate(38), dst_host_srv_serror_rate(39), dst_host_rerror_rate(40), dst_host_srv_rerror_rate(41)

Table 1. Type Of Features

Feature selection is the process of extracting features from the original data set because of irrelevant and redundant features can include noisy data it further affecting the accuracy of classification negatively by having the features with enough amount of information helps us to predict the results with high accuracy,due to the large amount of data processing and pattern recognition it's quite difficult for graphical representation sometimes it's not possible, so PCA becomes a powerful technique for data mining. Another advantage of PCA is after finding the patterns data gets compressed, using dimensionality reduction, without any loss of information [3].

Using weka tool found the following analysis after applying PCA.

=== Attribute Selection on all input data ===

Search Method:
Attribute ranking.

Attribute Evaluator (unsupervised):
Principal Component Attribute Transformer

Eigen value	proportion	cumulative	
7.37354	0.29494	0.29494	0.34129+0.32334-0.30238-0.30239-0.30125...
5.16652	0.20666	0.5016	0.40928+0.40927+0.40741+0.39840-0.22426...
2.06746	0.0827	0.5843	-0.5592-0.48236-0.35324-0.30237-0.25523...
1.87995	0.0752	0.6595	0.52824+0.38623-0.37337+0.35932-0.31431...
1.43559	0.05742	0.71692	0.43935-0.33537+0.3351+0.32930+0.3153...
1.1183	0.04473	0.76165	0.6171-0.50930+0.4685+0.1486-0.14135...

Table 2. PCA Analysis

Principal component analysis (PCA) is a technique that performs an orthogonal transformation for the data set of observations of possibly correlated features into a set of values of linearly uncorrelated features called principal components. This representation is defined in such a way that the first principal component has the largest possible variance with much of the variability in the dataset as possible, and each succeeding feature in turn has the large variance as possible under the constraint that is orthogonal to the remaining feature components.

PCA is sensitive along with relative scaling of the original dataset, Above PCA analysis is the result of weka tool for our dataset. After performing PCA to dataset choosing the attribute which is having high Eigen value in the overall attribute vector and then one with high variance and uncorrelated feature are extracted from the original dataset. Below figure depicts the variations in the attributes of working dataset.

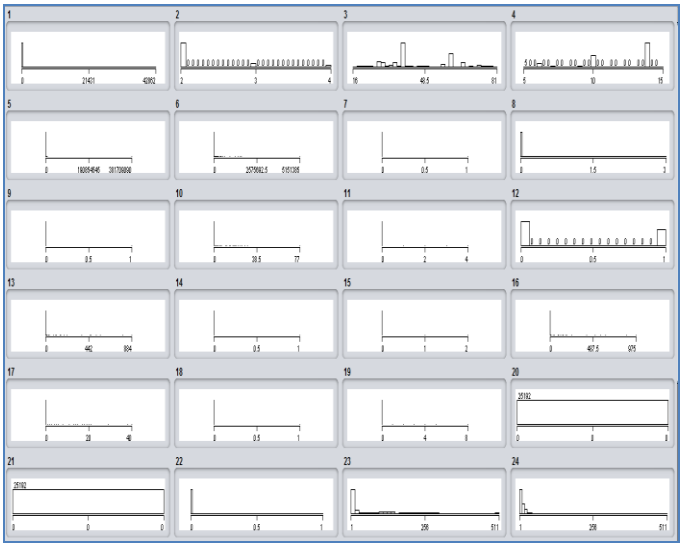

Figure 2. Snapshot of Variation in a attributes

IV. Scaled Conjugate Gradient Back propagation Learning Algorithm

In this algorithm we are using three layer feed-forward neural network along with sigmoid function at the hidden layer and softmax function at the output layer, It can classify the vectors appropriately by giving enough number of neurons at the hidden layer, and this network will be trained with Scaled Conjugate Gradient Backpropogation Learning Algorithm.

Neuron Model

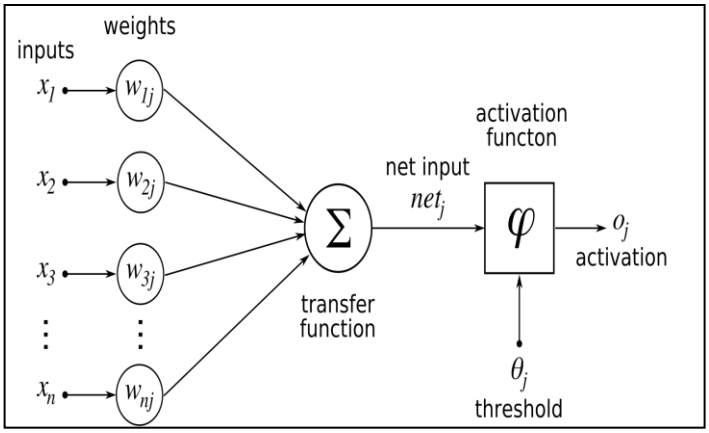

Figure 3 Basic Structure of Neuron

An elementary neuron with *n* inputs as each attribute in a dataset is shown above. Each input is weighted with an appropriate weight *w*. The sum of the weighted inputs along with bias forms the input to the transfer function $\sum$ for the next layer. These Neurons can use any transfer function to generate the corresponding output. Scaled Conjugate Gradient (SCG) is a supervised way of back propagation learning algorithm with feed forward neural networks, and it belongs to the class of conjugate gradient methods.

Scaled Conjugate Gradient Back propagation Learning Algorithm

step 1: Normalize the inputs and outputs with respect to their maximum values. Neural networks work better if input and outputs lies between 0 to 1 for each training pair is in a normalized form.

step 2: User can specify the number of neurons at the hidden layer based on features and dataset you can increase the number of neurons.

step 3: weights of synapses connecting between input neurons to hidden neurons represents as V_i and weights of synapses connecting hidden neurons to output neurons represents as W_i.

step 4: Compute the inputs to the hidden layer by multiplying corresponding weights of synapses.

step 5 : Let the hidden layer units evaluate the output using the sigmoid function as$\{O\}H = \{- - -1/(1+e^{(-I_{Hi})}- - -\}$.

step 6 : Let the output layer units evaluate the output using softmax function as $\{O\}O = \{- - - ((e^{(i)}/ \Sigma_{k=1}{}^{k} e^{i}{}_{k}---\}$this is the network output.

step 7: output error is back propagating by updating the weights and threshold values.

Line Search

In optimization phase, we are using line search strategy is one of the two basic iterative approaches to find a local minima X*, for the objective function f: $R^n \rightarrow R$. and the other technique is trust region. This line search method first determines a descent direction along with that objective function f will be reduced and then calculate a step size in that direction to determine how far X* should move in that direction. Below steps shows how we achieved this technique

Line search algorithm in Conjugate Gradient methods:-

1. For each iteration set counter k=0, and make an initial assumption x_0 for the minimum
2. Repeat:
3. Evaluate the descent direction in the curve as P_k
4. Compute α_k to loosely minimize $h(\alpha)=f(X_k + \alpha P_k)$ over $\alpha \in R$
5. Update $X_{K+1} = X_k + \alpha_k P_k$,and K=K+1
6. Until $\| \delta f(X_k) \| <$ tolerance

Trust Region

Trust region is one of the statistical optimization techniques to denote the subset of the region as objective function which is approximated using a model function as a quadratic. If the objective function is found within the trust region then the region gets expanded, in the reverse, if the approximation is not good then the region is contrasted. Trust region technique is also called as restricted step method.

In our conjugate gradient algorithms used a line search as an optimization technique at each iteration. This line search technique is computationally expensive, and it requires because network response to all training inputs must be computed many times for each search. The scaled conjugate gradient algorithm, developed by Moller and it was designed to avoid this time consuming line search in the line search optimization technique.

Time Complexity of SCG

When comparing SCG to other algorithms like standard back propagation algorithm the number of epochs is not relevant, indeed one iteration in SCG needs the computation of two gradients, and it also involves one call to the error function. Moller defines a complexity unit (cu) to be equivalent to the complexity of one forward passing of all patterns in the training dataset. Then computing the error function costs to 1 cu while computing the gradient can be estimated to the cost of 3 cu. According to Moller's metric, one iteration of SCG is as complex as around 10^{-16} iterations of standard back propagation algorithm [2].

V. RESULTS AND ANALYSIS

Selected features are applied to three different machine learning methods to measure the accuracy of classification as follows:-

A. Tree Based Classifier

This classifier uses decision tree as a predictive model for a particular dataset with input vector represents the branches of the tree and output target represents the leaves of the tree, decision tree classifies based on some test condition at each internal node of a tree when it reaches leaf node it finally assigns the class labels of that leaf node as result.

Figure 4 Snapshot of Tree classifier confusion matrix

Above figure 4 shows the confusion matrix of tree classifier represents the number of samples misclassified verses number of samples correctly classified in this overall 25192 samples 49 normal class misclassified as attack class and 68 attack class misclassified as normal class gives the 99.5% accuracy of classification after applying Decision tree based classifier.

B. Logistic Regression

In statistics, logistic regression is a kind of regression model with dependent variable is categorical, this paper covers the case of a binary dependent variable that is, where the output have only two values 0 represents normal and 1 represents attack, the case where the dependent variable has more than two outcome as class labels can be analyzed in multinomial logistic regression and it is an example of a discrete choice model for a qualitative response.

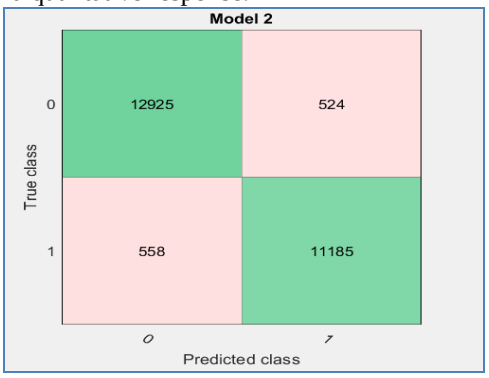

Figure 5 Snapshot of Logistic Regression classifier confusion matrix

figure 5 shows the confusion matrix of logistic regression classifier represents the number of samples misclassified verses number of samples correctly classified. In this overall 25192 samples 524 normal classes misclassified as attack class and 558 attack class misclassified as normal class. Gives the 95.7% accuracy of classification after applying Logistic Regression based classifier. Compare to tree its giving less accuracy because of number of dependent variable is very less in the chosen dataset.

C. Support Vector Machine (SVM)

Support vector machines are supervised way of machine learning models with learning algorithms involve support vector networks, it involves regression way classification process Given a set of training records, each marked as belonging to one or the other of two categories, an SVM algorithm builds a model that assigns new records to correct output class labels, making it a non-probabilistic binary linear classifier and this method uses Platt scaling as one line optimization technique. [8]

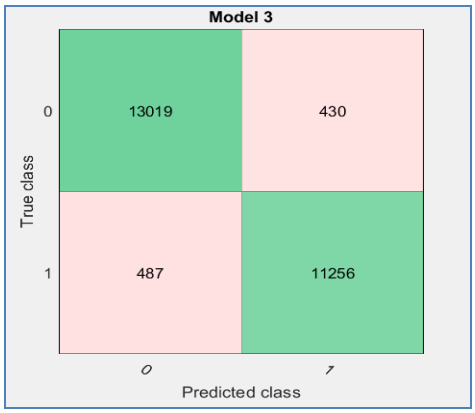

Figure 6 Snapshot of SVM classifier confusion matrix

Above figure 6 shows the confusion matrix of SVM classifier represents the number of samples misclassified verses number of samples correctly classified. in this overall 25192 samples 430 normal class misclassified as attack class and 487 attack class misclassified as normal class, gives the 96.4% accuracy of classification after applying SVM based classifier. Compare to logistic regression it is better SVMs can efficiently perform a non-linear classification using special technique called kernel trick, implicitly this mapping is achieved for the inputs of high-dimensional feature spaces.

The below figure shows results of Scaled Conjugate Gradient Backpropogation Learning Algorithm in terms of confusion matrices for training, validation, and testing, and the three kinds of data are combined. The network outputs has high approximation, as you can see by the high numbers of correct classification in the green squares and the less number of incorrect classification are in the red squares. And the lower right blue squares illustrate the overall accuracy.

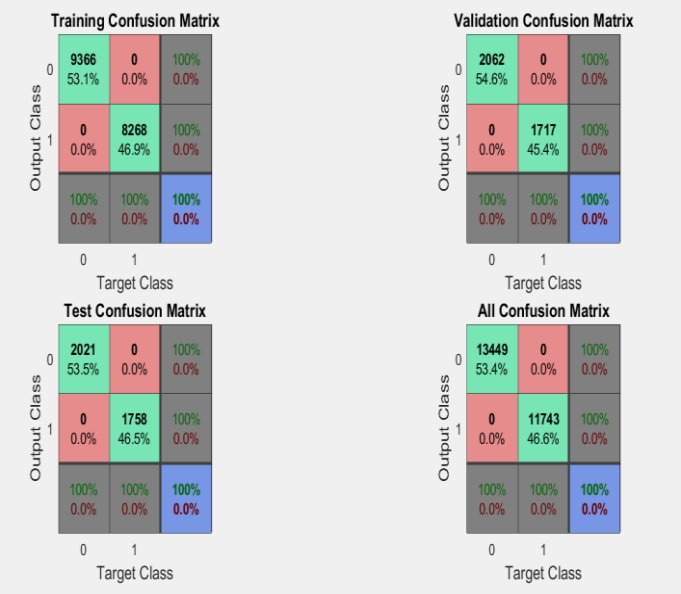

Figure 7 Snapshot of SCG confusion matrix of overall

In the overall complete dataset of 25192 instances using ANN its correctly classifies normal of class 0 with 13449 samples and attack of class 1 with 11743 samples overall it gives accuracy of classification as 100% with no false alarm rate.

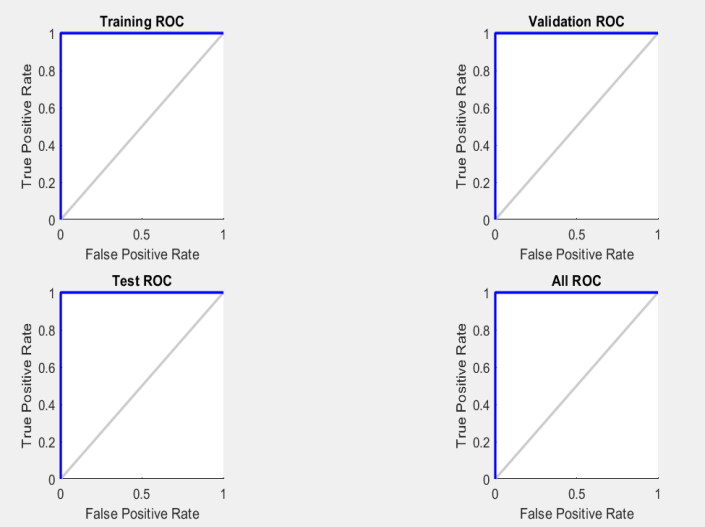

Figure 8 Snapshot of SCG ROC curve of overall

In statistics, a receiver operating characteristic curve, i.e. ROC curve, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied.

The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The true-positive rate is also known as sensitivity, recall or probability of detection in machine learning. The false-positive rate is also known as the fall-outer probability of false alarm and can be calculated as. The ROC curve is thus the sensitivity as a function of fall-out.

VI. CONCLUSION

A semi-automated network stream data mining technique classifies the NSL KDD dataset as binary classification of normal and attack packets by using scaled conjugate gradient back propagation learning algorithm which results in high accuracy of classification “Stream Data mining for Self Learning Network” using ANN this research is significantly revolutionized for challenges in the network security which eventually leads to excellent performance.

VII. FUTURE WORK

In the future focusing on Deep Neural Networks once in the traffic it identified that traffic leads to attack then classifying that traffic leads to what type of attack and how we can overcome with automatic network troubleshooting. So In the future planning to deploy deep learning technique as a predicting model for online streaming data and this work can be another high impact in the current research area.

References

[1] Vandana, M., and Sruthy Manmadhan. "Self learning network traffic classification." Innovations in Information, Embedded and Communication Systems (ICIIECS), 2015 International Conference on. IEEE, 2015.

[2] Aggarwal, Preeti, and Sudhir Kumar Sharma. "Analysis of KDD dataset attributes-class wise for intrusion detection." Procedia Computer Science 57 (2015): 842-851.

[3] Prof.Dighe Mohit S., Kharde Gayatri B., Mahadik Vrushali G., Gade Archana L., Bondre Namrata R , ”Using Artificial Neural Network Classification and Invention of Intrusion in Network Intrusion Detection System” International Journal of Innovative Research in Computer and Communication Engineering Vol. 3, Issue 2, February 2015.

[4] Palechor, Fabio Mendoza, et al. "Feature selection, learning metrics and dimension reduction in training and classification processes in intrusion detection systems." Journal of Theoretical and Applied Information Technology 82.2 (2015): 291.

[5] Bakhshi, Taimur, and Bogdan Ghita. "On Internet Traffic Classification: A Two-Phased Machine Learning Approach." Journal of Computer Networks and Communications 2016 (2016).

[6] Kumar, Sanjay, Ari Viinikainen, and Timo Hamalainen. "Machine learning classification model for Network based Intrusion Detection System." Internet Technology and Secured Transactions (ICITST), 2016 11th International Conference for. IEEE, 2016.

[7] Zhang, Zhuo, et al. "Proword: an unsupervised approach to protocol feature word extraction." INFOCOM, 2014 Proceeding IEEE, 2014.

[8] Jaiswal, Rupesh Chandrakant, and Shashikant D. Lokhande. "Machine learning based internet traffic recognition with statistical approach." India Conference (INDICON), 2013 Annual IEEE, 2013.