MINING NETWORK STREAM DATA FOR SELF LEARNING NETWORKS

Sunil Angadi
Dept. of Computer Science and Engineering,
Amrita School of Engineering, Bengaluru
Amrita Vishwa Vidyapeetham, India
Sunilangadi2@gmail.com

Radhakrishnan Gopalapillai
Dept. of Computer Science and Engineering,
Amrita School of Engineering, Bengaluru
Amrita Vishwa Vidyapeetham, India
g_radhakrishnan@blr.amrita.edu

***Abstract*— Detecting attacks on networks is a challenging task. Classifying packets as either normal packets or malicious packets is important to detect any attacks. This paper discusses methods to classify network traffic data as benign versus malicious. The original NSL-KDD dataset is classified using Artificial Neural Network. A taxonomy and survey of artificial neural network classification system is presented based on current and previous work. 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.**

***Keywords*— Artificial Neural Networks, IP Networks, Supervised Learning, Back propagation, Support vector machines**

I. Introduction

The network traffic is heterogeneous and it has traffic of variety of applications and utilities. These applications are different and there is a need to have 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. Meeting these requirements in a normal Local Area Network (LAN) that has good amount of bandwidth might be easy, but when considering internet connection with Wide Area Network(WAN), it’s 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 [1]. All existing classification methods based on source IP address, destination IP address and IP protocol, etc. has limited ability as the inspection is confined to the IP Networks of header only. Similarly, classification based on transport layer ports also has limitations. The problem with port based classification approach is that some applications may not be using standard ports or registered ports. Some applications themselves use well defined ports of other applications [2].

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 [3].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

An Artificial Neural Network (ANN) uses a collection of connected units called artificial neurons. These algorithms are inspired by biological neurons in a human brain. Synapse connects the neurons and they transfer a signal from one neuron to another neuron. Neurons that receive the signals, process the signals and then forward them to the neurons connected downstream. In ANN implementation, the synapse signal is a value, and it is evaluated by some transfer function on the sum of all its input value. Synapses and neurons also have a weight that varies as learning proceeds, which can decrease or increase the strength of the signal that it forwards to the next. Further, it has a threshold such that only if the overall output signals is below or above that level then downstream or upstream signal sent respectively. These neurons are arranged in layers [4]. Each layer may use different functions of alterations on their inputs and their corresponding signals propagate from the input layer to output layer, by moving through successive hidden layers.

II. RELATED WORK

Many of the researchers are working on the application of Machine Learning (ML) techniques with Artificial Intelligence discipline for internet network 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 intent can be predicted [5]. Features 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 train the designed machine learning model to predict this particular internet traffic as malicious.

These algorithms have a unique approach for selecting and assigning a priority set of features, which eventually gives 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 [6].

A. Network traffic classification using port based

In this method fixed registered port numbers are used for network traffic classification. It gives flexible approach to multiplexing of multiple flows to classify the traffic at the final application level. When client requests for the server, then server uses standard registered ports for the client response. If server uses well known port number, then port based traffic classification method works very well in all cases. The major drawback of this approach is new applications and protocols don’t follow standard registered ports and hence the prediction error rate is high.

B. Network traffic classification using payload based

Traffic classification is the first step that identifies different applications and protocols that exist in a network.To enhance the reliability of the classification of registered port numbers, some researchers proposed payload based traffic classification. This method is based on classifying the traffic based on payload information in a packet. This is also called as deep packet inspection. In this content, the captured packets are analyzed in depth so that it can classify the traffic accurately in a timely discriminative manner. However, in real time traffic this method is able to classify up to 69% of the total bytes accurately [7] which has limited accuracy.

Though ANN has been used in the past, we are investigating methods to improve the classification accuracy of ANN by using efficient optimization technique such as line search for mining network stream data.

III. CLASSIFICATION USING ANN

Dataset is collected from the communication network and it is a series of TCP packets with various network traffic parameters. Data collection is done between the interval data flows from a source to target IP address following some predefined protocol with some specific network constraints. Each connection has about 100 bytes of information and it is classified as either attack or normal.

NSL-KDD dataset is used to classify the attacks on four attack categories such as Denial of Service [Dos], Probing, Remote to Local [R2L], and User to Root [U2R]. This dataset consists of 41 features of which 3 are nominal type, 6 are binary type, and remaining 32 features are numeric type.

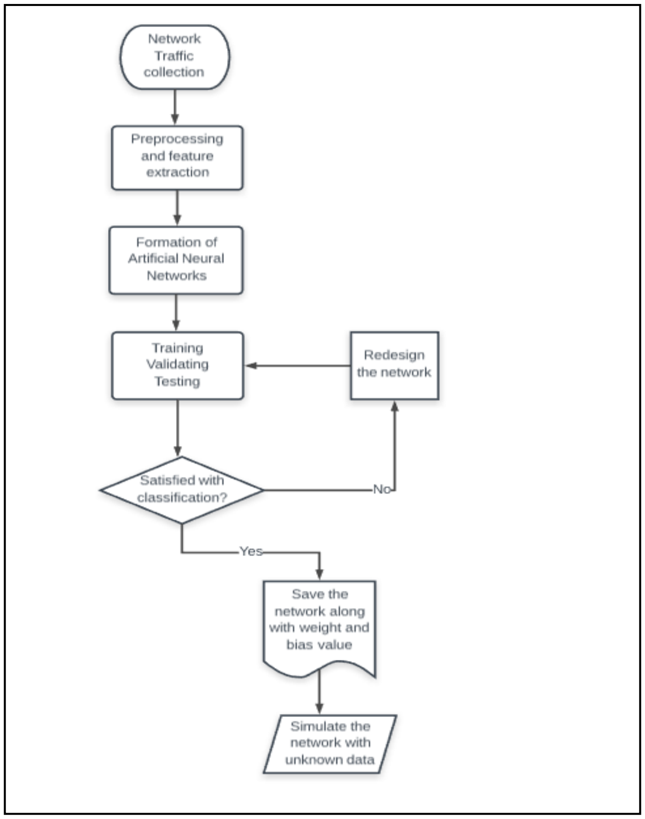
Network Traffic collection
Preprocessing and feature extraction
Formation of Artificial Neural Networks
Training Validating Testing
Redesign the network
Satisfied with classification?
No
Yes
Save the network along with weight and bias value
Simulate the network with unknown data

Fig. 1.Flow chart of ANN classification process

Figure 1 shows the flow chart of ANN classification in which for the NSL-KDD dataset is used. Important features in the dataset are identified using PCA (Principal Component Analysis). The selected features are trained in our model using ANN. The number of neurons in each hidden layer of ANN can be changed according to the input space. By varying the learning rate of neural networks in each epoch, the training is done repetitively till the model parameters achieve correct accuracy of classification. Once parameters are fixed then the model is tested with test dataset.

Feature selection is the process of extracting attributes from the working data set to remove irrelevant and redundant features. Noisy data can affect the accuracy of classification negatively. Having the features with right amount of information helps us to predict the results with high accuracy. Due to the large amount of data processing for pattern recognition it's quite difficult it's not impossible to graphically represent data, PCA becomes a powerful technique for feature extraction in these situations. Another advantage of PCA is after finding the patterns, data gets compressed, using dimensionality reduction, without any loss of information [8].

Weka tool was used to apply PCA and the results are shown in Table 1. All attributes were chosen for PCA using search method as attribute ranking and performing supervised attribute evaluator as principal component attribute transformer technique.

TABLE 1. PCA Analysis

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...

After performing PCA to dataset, the attribute which is having high Eigen value in the overall attribute vector is chosen and then one with high variance and uncorrelated feature are extracted from the original dataset [9]. Figure 2 depicts the variations in the attributes of working dataset.

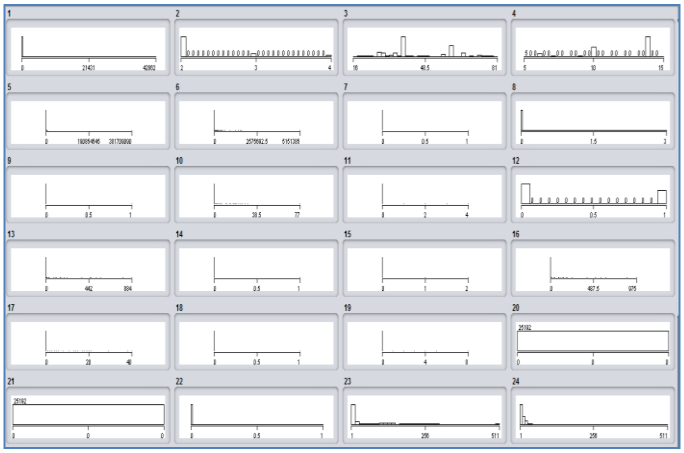

Fig. 2.Snapshot of Variations in Attributes

IV. SCALED CONJUGATE GRADIENT BACK PROPAGATION LEARNING ALGORITHM

In this algorithm, we are using a 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 inputs appropriately by giving enough number of neurons at the hidden layer then the network will be trained with Scaled Conjugate Gradient Backpropogation Learning Algorithm.

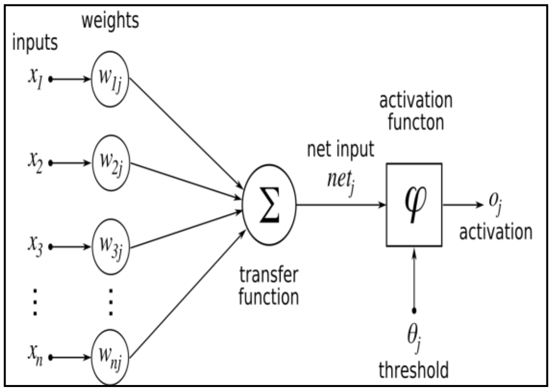

Fig. 3 Basic Structure of Neuron

Figure 3 shows the basic structure of a neuron with *n* inputs as each attribute in a dataset every input is weighted with proper weight *w* and sums of the weighted inputs along with bias forms the input to the transfer function ∑ for the next layer. These Neurons use sigmoid transfer function to get the corresponding output. Scaled Conjugate Gradient (SCG) is a supervised learning type of back propagation learning algorithm with feed forward neural networks, and it belongs to the class of conjugate gradient methods [10].

Algorithm 1 Scaled Conjugate Gradient Back Propagation Learning Algorithm

1. Normalize the input feature values to their maximum values using min-max normalization technique. A neural network performs better if input lies between the range of 0 to 1.
2. User can specify the enough number of neurons at the hidden layer based on features and dataset you can increase the number of neurons.
3. Input feature values represents as V_i and initialized weights for the neurons represents as W_i and bias value as b_i
4. Inputs to the hidden layer is calculated by multiplying the weights with input values along with adding neural network constants.
5. In the hidden layer each neuron units calculates the output using the sigmoid function as $\{O\}_H = \{- - -1/(1+e^{(-I_{Hi})})- - -\}$.
6. In the output layer each neuron units calculates the output using softmax function as $\{O\}_O = \{- - - ((e^{(i)}/ \Sigma_{k=1}^{k} e^{i}_{k}---\}$this is the network output.
7. Output error is back propagated by updating the weights, bias and threshold values.

A. Line Search

In optimization phase, we are using line search strategy as 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

Algorithm 2 Line search algorithm

```
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 $\alpha_k$ to loosely minimize $h(\alpha)=f(X_k + \alpha P_k)$ over $\alpha$€R
5.    Update $X_{K+1} = X_k + \alpha_k P_k$ ,and K=K+1
6.    Until $\| \delta f(X_k) \| <$ tolerance
```

B. Trust Region

Trust region is one of the statistical optimization techniques to denote the subset of the space as objective function that's approximated usage of model function as quadratic. If the objective function is found within the trust space 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.

V. Results and Analysis

The dataset is classified using ANN and three other machine learning methods to measure the accuracy of classification as follows:-

A. Classification Using 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 the objects based on some test condition at each internal node of a tree and when it reaches leaf node, it finally assigns the class label of that leaf node as predicted class.

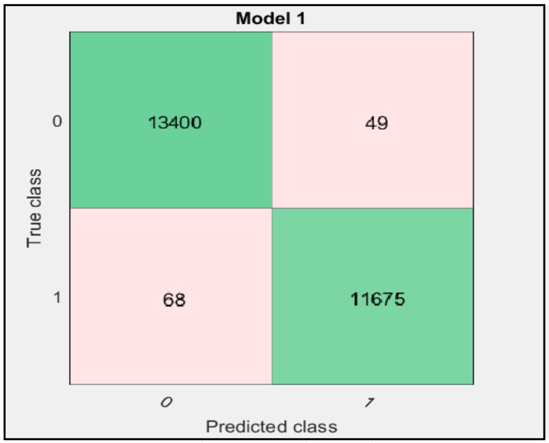

Fig. 4 Snapshot of Tree classifier confusion matrix

Figure 4 depicts the confusion matrix of tree classifier that represents the number of samples misclassified verses number of samples correctly classified. Out of a total number of 25192 samples classified, 49 samples belonging to normal class are misclassified as attack class and 68 samples of attack class are misclassified as normal class. Decision tree based classifier thus gave 99.5% accuracy.

B. Classification using Logistic Regression

Logistic regression is a kind of regression model where the dependent variable is categorical. Our problem focuses the case of a binary dependent variable where the result has only two values with 0 representing normal and 1 representing attack. The case where the dependent variable has more than two outcomes as class labels can be analyzed in multinomial logistic regression and it is an example of a discrete choice model for a qualitative response.

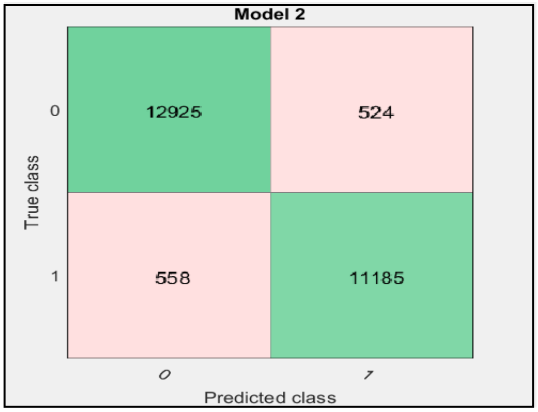

Fig. 5 Snapshot of Logistic Regression classifier confusion matrix

Figure 5 depicts the confusion matrix of logistic regression classifier. In this, 524 samples belonging to normal classes are misclassified as attack class and 558 samples belonging to attack class are misclassified as normal class. This gives 95.7% accuracy of classification for Logistic Regression based classifier. Compared to tree based classifier, it is giving less accuracy because of number of dependent variable is very less in the chosen dataset.

C. Classification using Support Vector Machine

Support vector machines (SVM) involve support vector networks. It involves regression classification process using a non-probabilistic methods. SVM uses Platt scaling as one line optimization technique.

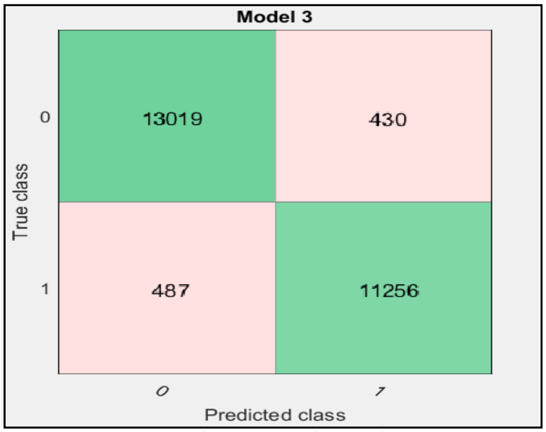

Fig. 6 Snapshot of SVM classifier confusion matrix

Figure 6 depicts the confusion matrix of SVM classifier. In this method, 430 normal class samples are misclassified as attack class and 487 attack class samples are misclassified as normal class, giving 96.4% accuracy of classification. Compared to logistic regression SVM is better because it can achieve better results in a non-linear classification using special technique called kernel trick. Implicitly this mapping is achieved for the inputs of high-dimensional feature spaces.

D. Classification using ANN

Figure 7 shows results of Scaled Conjugate Gradient Back propagation Learning Algorithm in terms of confusion matrices for training, validation, and testing, and the three kinds of data are combined. Our network output shows high approximation. It can be seen by the high numbers of correct classification in the green squares and the less number of incorrect classifications in the red squares. The lower right blue squares show the overall accuracy.

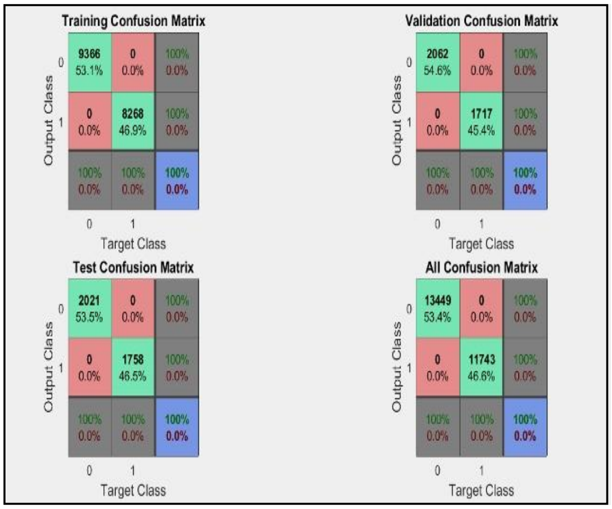

Fig. 7 Snapshot of SCG confusion matrix of overall

ANN correctly classifies all the 13449 samples with normal class as 0 and 11743 samples with attack class as 1 which gives an overall accuracy of classification as 100% with no false alarm rate.

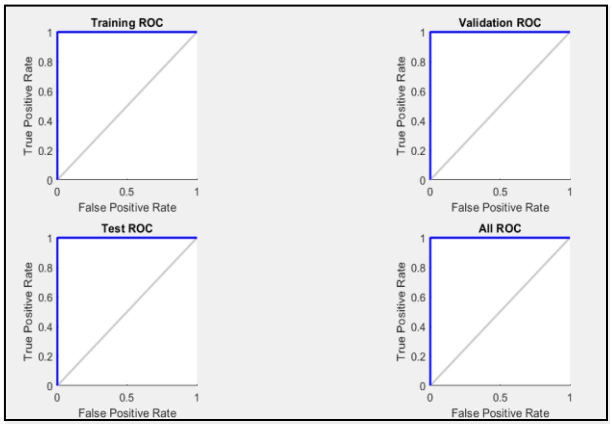

Fig. 8 Snapshot of SCG ROC curve of overall

The prediction results of binary classification model are analyzed with ROC (Receiver Operating Characteristics) curve. Figure 8 shows the ROC curve for training, validation, testing along with the overall results. From the ROC curve it can be concluded that our ANN prediction model is able to do binary classification with no false alarm rate.

VI. CONCLUSION

This study has taken NSL KDD dataset and classified the samples in to binary classes of normal and attack packets using four methods namely tree based classifier, logistic regression, support vector machines and Artificial Neural Network. The semi-automated network stream data mining technique using scaled conjugate gradient back propagation learning algorithm gives high accuracy of classification compared to other methods.

VII. FUTURE WORK

Deep Neural Networks methods may be tried out on more complex network data. Once it is identified that the traffic leads to attack then classifying that traffic in to the type of attack and how we can overcome with automatic network troubleshooting is possible.

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] Archanaa, R., et al. "A comparative performance analysis on network traffic classification using supervised learning algorithms." Advanced Computing and Communication Systems (ICACCS), 2017 4th International Conference on. IEEE, 2017.

[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.

[9] Malathi, A., J. Amudha, and Puneeth Narayana. "A Prototype to Detect Anomalies Using Machine Learning Algorithms and Deep Neural Network." *Computational Vision and Bio Inspired Computing*. Springer, Cham, 2018. 1084-1094.

[10] Rakesh, N. "Performance analysis of anomaly detection of different IoT datasets using cloud micro services." *Inventive Computation Technologies (ICICT), International Conference on*. Vol. 3. IEEE, 2016.