Artificial Neural Network based Intrusion Detection System: A Survey

Bhavin Shah
Associate Professor, MCA Programme
L. J. Institute of Management Studies
Ahmedabad, India.

Bhushan H Trivedi, PhD
Director
GLS Institute of Computer Technology
Ahmedabad, India.

ABSTRACT
Detecting unknown or modified attacks is one of the recent challenges in the field of IDS. Anomaly based IDS can play a very important role in this case. In the first part of this paper, we will focus on how ANN is recently used to address these issues. Number of the researchers has already shown the importance of the various Artificial Neural Network (ANN) based techniques for anomaly detection. In this paper, we will focus on Simple and Hybrid ANN based approach for anomaly detection. In simple approach we will discuss on how Back Propagation Neural Network (BPNN), Self Organizing Maps (SOM), Support Vector Machine (SVM), and Simulated Annealing Neural Network (SA) are used for anomaly detection? While in hybrid approach, we will focus on how more than one above technique are used? In the second part of the paper, we will try to compare the different ANN based techniques in terms of training time, number of the epochs required, converge rate, detection rate, learning approach, etc. Finally we will provide guidelines for the future work.

General Terms
Network Security, Intrusion Detection System, Anomaly based Intrusion Detection System, Artificial Neural Network.

Keywords
Intrusion Detection System (IDS), Anomaly Detection, Artificial Neural Network (ANN).

1. INTRODUCTION
As per detection technique, IDS can be classified in Signature Based Detection (or Misused Detection) and Anomaly Detection [6][9][10][11][12][13][14]. In case of Anomaly Detection, we are supposed to find the unusual behavior or abnormal activities in the network. From the historical data, we can generate the normal behavior of the system. Here, chances are high for the false alarm due to the various reasons like seemingly abnormal but actually normal behavior of the user itself. For example user genuinely changing a system file, wrong data for the normal behavior, 3 failed logins in a day as abnormal, but some users actually do so normally, and many more. Recent challenges in the field of IDS are to find out the Zero Day Attack and the Attack with Modified or Changed Behavior. Due to the self learning ability, ANN can plan very important role to address these issues. Following are some of the advantages and disadvantages of ANN.

Advantages of ANN:
1. It has self learning capability.
2. Performs tasks that a linear program can not.
3. When an element of the neural network fails, it can continue without any problem due to their parallel nature.
4. A neural network learns and does not need to be reprogrammed.

Disadvantages of ANN:
1. ANN needs training to operate.
2. The architecture of ANN is different from the architecture of microprocessors, therefore needs to be emulated.
3. Requires high processing time for large neural networks.

There are various types of the ANNs like: Back Propagation Neural Network (BPNN), Self Organizing Maps (SOM), Support Vector Machine (SVM), Radial Basis Function (RBF), Simulated Annealing Neural Network (SA) etc. To avail the advantages of more than one ANN techniques, researchers are using combination of the more than one technique (multi layer approach). In this paper, we will discuss IDS on both approaches: Simple ANN Based IDS, and Hybrid ANN Based IDS.

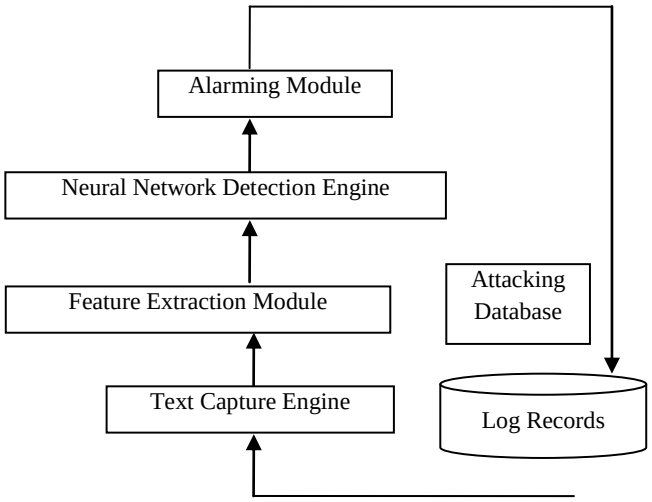
2. RELATED WORK
On the basis of the number of the ANN techniques used, ANN based IDS can be categorized as: 1) Simple ANN Based IDS and 2) Hybrid ANN Based IDS. In the first part of this section, we will focus on how Simple ANN Based IDS is recently used, while in second part, we will focus on Hybrid ANN Based IDS. In third part of this section, we will discuss about parameters or criteria, which can affect the performance in terms of the training time, number of the epochs required, converge rate, detection rate, learning approach.

2.1 Simple ANN Based IDS
In the simple ANN based IDS, any one ANN technique, like BPNN, KNN, SOM, SA is used. In this section, we will discuss one research paper to describe, how each of them is recently used in field of anomaly detection.

2.1.1 Intrusion Detection Systems Design based on Back Propagation Neural Network (BPNN) [2].
Among the researchers working on anomaly detection, BPNN is the first choice due to the number of the advantages over the other ANN techniques. Hence, authors of this paper used BPNN due to its ability of accurate prediction and better persistence.

As per the Figure-1, neural network classification engine distinguish the intrusive action by analyzing and processing the data given by the feature extraction module. If it finds it as aggressive behavior, it sends a warning message to the user,

and records the attack-related information, and updates the attack database for re-learning of neural network classification engine.



Authors, in their paper, had discussed the following key issues for designing the BPNN.

Fig 1: Zhang Wei, Wang Et al. BPNN Model

1) Selection of layers: As per the authors, complexity of the problem is not very high. They had used only one hidden layer and due to this, efficiency of the system is good.
2) Determination of number of input and output layer: The dimension of BPNN's input and output layer may depend on actual request. Authors had used KDD CUP 1999 dataset [15]. Out of the 41 inputs of the dataset, authors had used 8 important features as input and 5 as output.
3) Determination of number of the neurons in the hidden layer: If the nodes in the hidden layer are very few, the network non-linear mapping function and fault tolerance would be very poor. If too many, learning time would increase, and learning error is not necessarily the best. When the numbers of the samples for input training are large, the hidden layer nodes are concerned not only to the sample number, but also to the volatility of the approximating function. With the increase of the number of samples and the enlargement of volatility of the approximation function, the hidden layer nodes should be increased accordingly. But, when the complexity of network will increase, the network convergence rate will be slow. So the scope of network cannot arbitrarily enlarge.

2.1.2 Intrusion Detection: Support Vector Machines and Neural Network [7].

Authors used KDD CUP 1999 Data set for training and testing their model. Data were classified in to two classes: Normal (+1) and Attack (-1). They had used the SVM light freeware package. For data reduction, they had applied SVMs to identify the most significant features for detecting attack patterns. The procedure is to delete one feature at a time, and train SVMs with the same data set. By this process, 13 out of the 41 features of KDD CUP 1999 dataset [15] are identified as most significant: 1, 2, 3, 5, 6, 9, 23, 24, 29, 32, 33, 34, and 36.

Authors composed two training sets containing the same 10000 data points, with respectively 41 features and 13 features each. The 10000 data points were randomly generated, which include a subset of data points from each of the 23 classes in proportion to their size. Data points were randomly generated and contain actual attacks and normal usage patterns. Training was done using the RBF (Radial Bias Function) kernel option. In their experiment, authors got 98.9% accuracy for true negative case, and 99.7% accuracy for true positive case.

2.1.3 Intrusion IDS using Self Organizing Maps [1].

Authors use this model, to transform an input data set of arbitrary dimension to a one- or two-dimensional topological map. The structure of the SOM is a single feed forward network, where each source node of the input layer is connected to all output neurons. SOM model is shown in Figure-2.

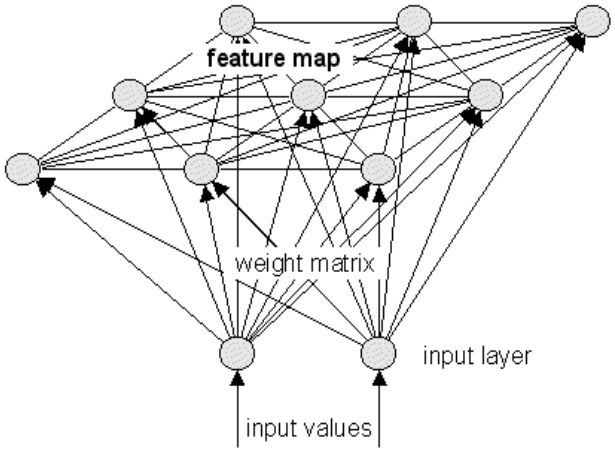


Fig 2: SOM Model

Following are the steps of the model, proposed by the authors.
1) One sample vector x is randomly drawn from the input data set and its similarity (distance) to the codebook vectors is computed by using Euclidean distance measure:

$$||x - m_e|| = {}^{min}_{t}\{||x - m_i||\} \quad (1)$$

2) After the Best Matching Unit (BMU) has been found, the codebook vectors are updated. The BMU and topological neighbors are moved closer to the input vector in the input space i.e. the input vector attracts them. The magnitude of this attraction is governed by the learning rate. As the learning proceeds and new input vectors are given to the map, the learning rate gradually decreases to zero. Along with the learning rate, the neighborhood radius also decreases. The update rule, for the reference vector of unit i, is given by:

$$m_i(t+1) = \begin{cases} m_i(t) + \alpha(t)[x(t) - m_i(t)], t \epsilon n_c(t) \\ m_i(t), i! \epsilon n_c(t) \end{cases} \quad (2)$$

3) The steps 1 and 2 together constitute a single training step and they are repeated until the training ends. The number of training steps must be fixed prior to training.

After the training is over, the map should be topologically ordered. It means that, n topologically close input data vectors map to n adjacent map neurons or even to the same single neuron.

For the training purpose, authors has constructed 30 X 30 SOM map and used DARPA dataset [16]. While for collecting the data from the real traffic, they developed their own packet sniffer. Authors used batch training algorithm with training length 100 and starting radius 15. Self organizing map was largely successful in classifying the IP packets in three classes: intrusion, possible intrusion or normal.

2.1.4 Improved Simulated Annealing Neural Network [5].

The Simulated Annealing Neural Network (SA) uses heuristic random searching method. SA not only accepts the solution which makes the objective function value get "better", but can also accepts the solution which makes the objective function value get "worse" at a definite probability. With temperature reduction, the accepting probability will gradually decrease. Due to this, it can avoid the local optimum solution and get a global optimum solution. Simulated Annealing needs lots of iterative computation to train the neural network which leads to the slow convergence rate.

To get the higher accuracy and fast convergence rate, authors had used Improved Simulated Annealing Neural Network. They used Powell algorithm to form improved simulated annealing mixed optimize algorithm, instead of gradient falling algorithm of BP network.

Authors, for their experiment, set initial temperature T =30, the iteration times as 50, the learning efficiency as 0.8, the inertia coefficient as 0.7, the system error as 0.01, the maximum error of single sample as 0.001, the iteration time of network as 10000. They used DARPA 1999 training dataset [16].

2.2 Hybrid ANN Based IDS

In hybrid ANN based IDS, more then one ANN techniques are implemented one after another. During our study, we came across various model which had implemented one ANN techniques followed by other. In this paper, we will discuss some of them.

2.2.1 Octopus-IIDS: An Anomaly Based Intelligent Intrusion Detection System (IIDS) [3]

Main objective of octopus-IIDS is to provide an intelligent intrusion detection system (IIDS) that is accurate, flexible, tolerant to variations of attacks, adaptive to changes in the network, modular and that operates in real time. Authors proposed two layers approach.

First layer (Classifier Layer / KNN Layer): This layer has been used to reduce the false negative rate. It analyzes and classifies the network traffic into (DOS, Probe, R2L and U2R) present in the KDD CUP 1999 Dataset. Even normal traffic is also classified in to it. Kohonen Neural Network (KNN) has been used for this separation, as it supports unsupervised learning. It can separate known patterns, generalize the patterns, and detect variations of attacks also. In this model, KNN contains 41 inputs and 4 outputs. The output of this layer is given to the specialized classifier layer (SVM Layer).

Second Layer (Decision Layer / SVM Layer): Support Vector Machines (SVM) can be used to improve the detection rate. Here, data/traffic is separated into two classes: 1) Normal traffic and 2) Malicious traffic. Authors used SVM because of two reasons: (1) In the identification of the anomalies, SVM is more efficient [8]. (2)SVM can bear a certain amount of noise in the input. In selection of configuration parameters, SVM networks are less complex than other neural network models due to the number of hidden layers, number of nodes for each layer and transfer functions. The wrong choice of some of these parameters may cause degradation in performance of the network. In octopus model, detection ratio is very good in KDD as well as Real Data, which can be seen from the Table 1.

Table 1. Result of Octopus IDS

Data Set	Detection Rate
KDD CUP	97.40% with maximum deviation of 8.57%
Real Data	83.90%with maximum deviation of 9.72%

2.2.2 The Research of Dynamic Change Learning Rate Strategy in BP Neural Network and Application in Network Intrusion Detection[4].

Objective of this paper is to reduce the training steps. Authors presented two layers approach.

1) The strategy of dynamic change learning rate in Back Propagation Neural Network (BPNN**):** Performance of the BPN depends upon the learning rate η. If η is taken as a constant, then it will bring to the local minimum and slow convergence rate. Small value of η makes the training times increase in flat area of the error surface, while the big value of η also leads to increase training time in gradient area. So, authors suggested: to initialize the step value η firstly, if the error increases after time iteration, this iteration is prove to be invalid. So the step value multiplies with a value of β (<1), as η (t+1) =βη(t), and repeats iteration again. If the error decreases, this iteration is prove to be valid, so the learning rate multiplies with a value of α (>1), as η (t+1) = αη(t) . This method improves the slow convergence rate. However, learning rate need to be adjusted when the system error of this iteration isn't up to the expectation. Authors have suggested the following formula for adjustment rule of learning rate in BPNN.

$$\eta(n) = \begin{cases} \frac{E(n-1)}{E(n)}\,\eta(n-1) & R_{min} \le \frac{E(n-1)}{E(n)} \le R_{max} \\ R_{min} * \eta & \frac{E(n-1)}{E(n)} < R_{min} \\ R_{max} * \eta & \frac{E(n-1)}{E(n)} > R_{max} \end{cases} \quad (4)$$

2) Simulated annealing algorithm: Simulated Annealing (SA) can help the BPNN to avoid from the local minimum. Here, the alterable weight values of net equal to the metallic particle, while the output error equals to the energy state of metal. Authors use the eight steps to optimize BP network using SA algorithm.

Authors had implemented the proposed system with MATLAB 7.0 by using KDD KUP dataset. As per the results shown in Table 2, Dynamic Change BPNN takes very less training steps as compare to other BPNN.

Table 2. Comparison of Different BPNN techniques based on training steps

Data Set	Training Steps
Standard BP algorithm	28000
Changing step length of BP	2544 maximum
Dynamic Change learning rate in BP	1800

2.3 Discussion on Different ANN Models

BPNN : BPNN is supervised learning method in which set of the input and expected output must be provided. BPNN can have one or multiple hidden layers. Optimal number of the hidden units for given number of inputs and outputs, can be decided by the trial and error method. For the problems like detecting and categorizing the attack, one hidden layer is more than sufficient. Even two or three hidden layer can be implemented but it will increase the complexity of the system and hence will reduce the convergence rate. Number of the units in each layer can also increase or decrease the complexity of the system. Too much hidden units can reduce the performance of the system, while too low hidden units can reduce the detection rate.

We implemented the BPNN by using the KDD CUP 1999 dataset [15] of MIT Lincoln Laboratory. We took 10% (= 500000+ input rows) of the dataset as training and 90 %(= 40000000+ input rows) dataset for the testing. The dataset contains 24 types of training attacks, with an additional 14 types in the test data. All the attacks fall into four main categories: DOS, R2L, U2R and Probe attacks. We implemented two experiments. For both the experiments, we took 41 inputs, one output, learning rate as constant (0.9), and initial weights were set as random value. In our first experiment, we took two hidden layer with 41 hidden units in each layer. While in the second experiment, we took one hidden layer with 42 hidden units. Our experiments shows that model with one hidden layer takes less time for the training as compare to the two hidden layer. Even convergence rate is also high for the model with one hidden layer. During our both the experiments, we observed that BPNN is suffering from the local minima, and slow coverage. Performance is good in detection of the known and unknown attack. But, to train the BPNN, number of the epochs required was very high which lead to very high training time. If network is over trained then it can decrease the performance, and to overcome, one has to define the early stopping condition. As BPNN can support multiple output unit, it is possible to classify the given data record in to one of the attack category of KDD CUP.

SVM: SVM is supervised learning method which can able to classify the data into the binary form: Attack or Normal. It cannot be able to classify the attack data in to the specific category like DOS, R2L, U2R and Probe just like BPNN. Key feature of the SVM is the absence of local minima. In SVM, classification of the data is very faster, and real time performance is also very good. Number of the epochs required is not much high. Even for each epochs, training time is very low. We can train the SVN network with very less time as compare to the other ANN. SVMs are relatively insensitive to the number of data points and the classification complexity does not depend on the dimensionality of the feature space. So SVM can potentially learn a larger set of patterns and thus be able to scale better. SVM can bare noise in the input data. SVM has the high algorithmic complexity and require extensive memory for the implementation.

SOM: It is unsupervised algorithm that works with nonlinear data set. It has excellent capability to visualize high dimensional data onto 1 or 2 dimensional space, which makes it unique especially for dimensionality reduction. Self-organizing maps are different from other artificial neural networks in the sense that they use a neighborhood function to preserve the topological properties of the input space. In the SOM learning algorithm there is only one winning neuron, and due to this, there can be several nodes of the network which remains underutilized or completely unutilized. In SOM, weight adjustment is determined by learning rate and the difference between the input pattern and the winner neuron's weight. SOM ignores some correlative relationships during the learning phase, which actually exist between the input pattern and the weights of all the nodes that participate in competitions, which can affect the detection rate and also lead to lower stability of the IDS. It is time consuming algorithm, and performance depends upon the number of the neurons used. If number of the neuron increase, then computation increases. One major problem with SOM is getting the right data. Unfortunately one needs a value for each dimension of each member of samples in order to generate a map. In case of IDS, to get the data for each dimension is very difficult, so this is a limiting feature to the use of SOM often referred to as missing data. Another problem is that every SOM is different and finds different similarities among the sample vectors. SOM organize sample data so that in the final product, the samples are usually surrounded by similar samples, however similar samples are not always near to each other. So a lot of maps need to be constructed in order to get one final good map.

Improved simulated annealing: Improved simulated annealing neural network has fast convergence and has the characteristic of global approximation. Due to its ability of self adjusting and adapting the weight values and threshold values, it enhances the training accuracy and speed.

Octopus Model (SOM +SVM) : As per the model, attack traffic as well as the normal traffic, must pass from both the layers. As the output of the second layer, the traffic will be classified as normal or as attack. Until that, all the traffic will be treated as same. So, for normal traffic, high processing is require, which can affect the performance of network. As per the Authors, in comparison with other systems having one layer approach, the overhead of their system is high by 60%. Their prototype model gives 83% success rate when they tested it with the real traffic. This result is good but still, there is a scope for the improvement.

Dynamic Change Learning Rate BPNN With SA: In this model, we need to set the initial temperature high enough to ensure that we won't trap in a local minimum. It means that the value of η must be high in BPNN. High value of η leads to increase training time in gradient area.

By using the combine approach of BP and SA, we can reduce the training steps, but as the simulated annealing algorithm has been used, the number of interactions required for each training steps is very high [5]. As this is two layer approaches, the performance of the system in real time will low as compare to the one layer system.

3. COMPARISON OF DIFFERENT ANN MODELS

In this section, we will try to compare different ANN models presented in this paper on the basis of the factors like: objective of the method, learning approach (supervised learning or unsupervised leaning), detection rate, training time, convergence rate, attack classification, dataset used and

overhead if any. It should be noted that for each and every methods, there are various parameters and criteria (like in BPNN :number of hidden layer, number of the units in the hidden layer, learning rate, number of the inputs, number of the outputs, over fitting, over training) which can play very important role in the performance of the model. We found that the papers we had discussed here, do not have these parameters as same. So, statistical comparison is not possible and hence, we have to make non statically comparison.

Objective of Model: As per the [2], objective of the BPNN is to get more persistence and accurate prediction, while SOM can be useful for novelty detection, automated clustering and visual organization [1]. As per the [7], SVM is probabilistic binary linear classifier, which can easily scalable to more data points and can be useful for the efficient and highly accurate prediction. While SA is more suitable for the higher accuracy as well as for faster convergence speed[5]. Hybrid ANN based model can be useful when high detection rate is required. As per the authors of [4], Dynamic Change BPNN with SA will improve the detection efficiency, real-time property and also reduces the training time and training cost. While in Octopus IDS [3], KNN and SVM has been used which can give accurate results. This model is more flexible, tolerant to variations of attacks, adaptive to changes in the network, modular and can operates in real time also.

Learning Approach (Supervised or Unsupervised) : Majority of the models discussed in this paper (Except SOM) uses supervised learning approach. Supervised approach is useful when training samples with their expected output, are available. With unsupervised learning just like in SOM, it is possible to learn larger and more complex models than with supervised learning. This is because in supervised learning one is trying to find the connection between two sets of observations. The difficulty of the learning task increases exponentially in the number of steps between the two sets. Hence, supervised learning cannot learn models with deep hierarchies. But in the case like anomaly detection, hierarchies are not too much high. In addition to this, standardized training data set KDD CUP 1999 is also available to train the ANN based IDS. So, supervised approach is widely used by all the authors except [1].

Detection Rate: When we compared the detection rate of the different model discussed in this paper, we found that most of the model have very good detection rate except SVM. Detection rate of BPNN is very good as compare to the other ANN based techniques. It should noted that detection rate of the any model depends upon the number of variable and other criteria. For every method, there are advancement is available. By implementing it, we can improve detection rate just like in [3], which is updated version of [1].

Training Time and Number of Epochs :Number of the epochs required to train the network is very high in case of BPNN, which is lowest in Dynamic Change Learning Rate BPNN with SA[4]. As per the octopus IDS[3], number of the epochs required is less. SVM and Dynamic Change Learning Rate BPNN with SA has very low training time, while BPNN and SOM requires very high training time.

Attack Classification: As SVM is a binary classifier, it cannot classify the attack in to the specific class. Other then SVM, all the other techniques discussed in this paper, can detect and classify the attack in to the specific class.

Response Time: Response time is very fast in case of SVM due to its inherent capability of binary classifier. Even improved SA[5] also has very fast response time as compare to the other techniques.

Can able to Detect new Attack: As all the techniques discussed in this paper, are Artificial Neural Network based, each is capable to detect new attack or modified attack.

Dataset Used : To train and testing the ANN models, very rich training data set is required. All the authors had used either KDD CUP 1999 dataset (The Third International Knowledge Discovery and Data Mining Tools Competition) [15] or DARPA dataset [16]. These both the datasets are standardized and freely available on the internet. KDD Dataset has 42 columns. Last column of this dataset is related with given row is either attack or normal. KDD Dataset contains approximately 500000+ rows for training and 40000000+ rows for the testing. As per the [17], [18], and [19] there are number of the limitations of these dataset. Now a days, NSL Dataset [20], which is improved version of the KDD CUP Data set is also widely used.

Overhead: In the case of Hybrid ANN base IDS, overhead is always there due to the one ANN model followed by the other ANN model. In octopus model [3], this overhead is 60%.

4. CONCLUSION

There are various techniques of Artificial Neural Network, which can be applied to Intrusion Detection System. Each technique is suitable for some specific situation. BPNN is easy to implement, supervised learning artificial neural network. Number of the epochs required to train the network is high as compare to the other ANN techniques. But, detection rate is very high. BPNN is suffering from the local minima and slow coverage. So, to improve the detection speed in real time, one hast to implement the techniques available like in [4]. BPNN can be used when one wants to not only detect the attack but also to classify the attack in to specific category so that preventive action can be taken. While SVM can be used for classification and regression analysis. SVM is a non-probabilistic binary linear classifier, which can classify the data in to the binary form only. It can only be useful to find out whether given data is attack or not? Due to its low training time and high detection efficiency, it can be useful for the critical and highly secure systems where we want quick decision about the data, whether it is attack or normal. If it is an attack, then issues like, which kind of attack, what action should be taken, can be decided latter on, but quick response is require.

SOM is useful for visualizing low-dimensional views of high-dimensional data. It is also effective for novelty detection, automated clustering and visual organization. Due to its inherent ability, it can be useful for the pattern discovery. Simulated annealing (SA) is a generic probabilistic global optimization problem of locating a good approximation to the global optimum of a given function in a large search space. It is often used when the search space is discrete. A simulated annealing algorithm searches for the optimum solution. Specifically, it moves about randomly in the solution space looking for a solution that minimizes the value of objective function.

Researchers are using simple or hybrid ANN based approach for detecting attack. Simple ANN approach is fast but not accurate, while multilayer (hybrid) approach is more accurate but not fast as more than one neural networks is associated with other ones. By combining the different ANN techniques just in [3] and [4], one can reduce the number of the epochs required and hence can reduce the training time.

5. FUTURE WORK

During our study we found that, factors like Detection Rate, Training Steps, Training Time, Performance with Real Data, Category of Attacks Detected, Throughput, and Overhead can play very important role while selecting the IDS. There are number of the systems developed which are focusing on one or other issues listed above. But, we didn't find any model which is focusing on all or majority of them. We are planning to develop the same kind of system in future.

6. REFERENCES

[1] V. K. Pachghare, Parag Kulkarni, Deven M. Nikam , 2009, Intrusion Detection System Using Self Organizing Maps, IEEE.

[2] Zhang Wei, Wang Hao-yu, 2010, Intrusive Detection Systems Design based on BP Neural Network, IEEE.

[3] Paulo M. Mafra, Vinicius Moll, Joni da Silva Fraga, 2010, Octopus-IIDS: An Anomaly Based Intelligent Intrusion Detection System, IEEE.

[4] Song Guangjun, Zhang Jialin, Sun Zhenlong, 2008, The Research of Dynamic Change Learning Rate Strategy in BP Neural Network and Application in Network Intrusion Detection, IEEE, and also in 3rd International Conference on Innovative Computing Information and Control (ICICIC'08).

[5] Meijuan Gao, Jingwen Tian, 2009, Network Intrusion Detection Method Based on Improved Simulated Annealing Neural Network, IEEE and also at International Conference on Measuring Technology and Mechatronics Automation.

[6] Jing Bi, Kun Zhang, Xiaojing Cheng , 2009, Intrusion Detection Based on RBF Neural Network, IEEE and also at International Symposium on Information Engineering and Electronic Commerce.

[7] Srinivas Mukkamala, Guadalupe Janoski, Andrew Sung, 2002, Intrusion Detection: Support Vector Machines and Neural Networks, IEEE and Proceedings of the 2002 International Joint Conference on Neural Networks IJCNN02 Cat No 02 CH3 7290.

[8] X. Haijun, P. Fang, W. Ling, and L. Hongwei, 2007, Ad hoc-based feature selection and support vector machine, IEEE and also at Grey Systems and Intelligent Services 2007 GSIS 07.

[9] Yingbing Yu, Anomaly Detection of Masqueraders Based Upon Typing Biometrics And Probabilistic Neural Network, , ACM and also at Journal of Computing Sciences in Colleges, Volume 25 Issue 5.

[10] Milan Tuba, Dusan Bulatovic, 2010, Design of an Intrusion Detection System Based on Bayesian Networks, ACM.

[11] Nabeel Younus Khan, Bilal Rauf, Kabeer Ahmed, 2010, Comparative Study of Intrusion Detection System and its Recovery mechanism, IEEE.

[12] Ondrej Linda, Todd Vollmer, Milos Manic, 2009, Neural Network Based Intrusion Detection System for Critical Infrastructures, IEEE and also at Proceedings of International Joint Conference on Neural Networks.

[13] P. Anderson, Computer security threat monitoring and surveillance, Technical report, James P. Anderson Co, 1980.

[14] D. E. Denning, 1987, An Intrusion Detection Model, IEEE Transactions on Software Engineering, Vol. SE-13, February 1987, pp. 222-232.

[15] KDD Cup 1999 Data [EB/DL], 1999, University of California, Irvine. http://kdd.rcs.uci.edu/databases/kddcup99/kddcup99.htm

[16] DARPA Intrusion Detection Evaluation Data Sets, 2002, MIT Lincoln Laboratory. http://www.ll.mit.edu/IST /ideval/data/data_index.html

[17] Matthew V. Mahoney and Philip K. Chan , 2003, An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection, Proceedings of the Sixth International Symposium on Recent Advances in Intrusion Detection, Springer.

[18] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani, 2009, A Detailed Analysis of the KDD CUP 99 Data Set, Second IEEE Symposium on Computational Intelligence for Securityand Defense Applications (CISDA) 2009.

[19] J. McHugh, 2000 , Testing intrusion detection systems: a critique of the 1998 and 1999 DAPRA intrusion detection system evaluations as performed by lincoln laboratory, ACM Transactions on Information and System Security, vol. 3, no. 4, pp. 262–294, 2000.

[20] NSL- KDD Dataset, Faculty of Computer Science University of New Brunswick, http://www.iscx.ca/NSL-KDD.