J. Basic. Appl. Sci. Res., 2(9)9542-9547, 2012

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

A New Approach for Improvement RBF/GRNN/PNN in Intrusion Detection

Maisam-Mohammadian[*1] Nasser-Mozayani[2] Mohammad Karimi[3]

[1]Department of Computer Eng., Iran University of Science and TechnologyNarmak, Tehran Iran
[2]Assistant Professor at Department of Computer Eng., Iran University of Science and Technology Narmak, Tehran,Iran
[3]Department of Computer Eng, Islamic Azad University Takestan,iran

ABSTRACT

As attacks became more complicated, the traditional methods such as firewalls were not successful in exact diagnosis. This problem could have harmful effects. This caused Intrusion detection system (IDS) to find an important role in network security. IDS can be categorized into two groups: misuse and abnormal detection. Misuse detection compares data to well-known attack signature so it cannot diagnose unknown attacks. Abnormal detection has better performance to detect new attacks by modeling. Attacks can be classified into four groups: DoS, Probe, U2R, and R2L. Many approaches have been used in IDS. One of them is artificial neural network (ANN). ANN has widely been used in IDS. Among all ANN types, RBF has shown better performance. This paper tries to implement a hybrid Artificial Neural Network in Intrusion Detection System based on RBF. This study investigates the effectiveness of RBF/GRNN/PNN on true diagnosis of attacks. The results are compared to support vector machine and self organizing map to evaluate simulation results. Our results show improvement compared to these two methods.
KEY WORD:IDS(intrusion detection system), DOS, Neural Network, RBF/GRNN/PNN

I. INTRODUCTION

Since the traditional prevention methods have failed to protect network completely, IDS now has find an important role in providing security.

IDS can be classified into two groups: misuse detection and anomaly detection. Misuse detection is done by comparing data to descriptions of intrusion behavior. So it cannot detect unknown attacks. In anomaly detection, normal behavior is modeled so abnormal behavior can be found out. Anomaly detection can be found out in two ways. One of them is static detection. In this method, it is assumed that behavior of monitored target never changes. Dynamic anomaly detection is the other one. It extracts data from usual habit behavior of users [1]. Anomaly detection can find new unknown attacks. IDS can have other characteristics: i) detection method which can be done by misuse or anomaly, ii) response to instruction which can be passive or active, iii) audit data source in hosts or networks, iv) location of detection that can be distributed or central [1].

Attacks fall into four main categories:

- DOS: denial-of-service, e.g. syn flood;
- R2L: Remote to local, unauthorized access from a remote machine, e.g. guessing password;
- U2R: User to root, unauthorized access to local super user (root) privileges, e.g., various ``buffer overflow'' attacks;
- Probing: surveillance and other probing, e.g., port scanning.

Up to now different approaches have been used in IDS. Artificial neural network is one of the most popular and effective one which will be discussed later. Fuzzy logic is another effective approach. It can make flexible models for anomaly and misuse detection. Another good approach is evolutionary computation. It can greatly be used in searching for optimal solutions, automatic model design, and classifiers to solve detection problems. Artificial immune systems can widely increase misuse and abnormal detection. Their attributes can help to have a dynamic, distributed, and self organized intrusion detection system [1]. Ant colony optimization and particle swarm intelligence have also acceptable performance in intrusion detection system. Among all these methods mentioned, artificial neural network can still be improved to have better performance in detection.

***Corresponding Author:** Maisam-Mohammadian ,Department of Computer Eng., Iran University of Science and TechnologyNarmak, Tehran Iran

An artificial neural network consists of neurons which are processing units. They can be classified into two groups: supervised learning, and unsupervised learning. When IDS was first developed, Multi-layered feed forward neural network back-propagation (MLFF-BP) was effectively used for anomaly detection. In some studies, information such as command sets, and login host addresses were used to distinguish normal and abnormal behavior while others considered patterns of commands or software behavior [2-5]. Redial basis function neural networks (RBF) are popular type of feed forward neural networks. They are faster than back propagation because they do classification by measuring distances between inputs and the centers of RBF hidden neurons. Until now different studies have been done on RBF. Previously a hierarchical RBF was proposed for misuse and anomaly detection [6]. In first layer, RBF anomaly detector decides an event is normal or not. Misuse RBF detector is done in second layer. Other studies showed that MLFF-BP is better than RBF for misuse detection but it is time consuming in training . For anomaly detection RBF has better performance [6,7].

Other studies have been done on other types of neural networks. Recurrent neural networks are one of tested approaches which has been tested [4,8,10,11,12]. These networks can be used to predict whether the event is an attack or not. They use memory for prediction. Unsupervised learning such as self-organizing maps (SOM) as well as supervised learning were also investigated [13]. SOMs are popular neural network for anomaly detection [14-16]. It has also been tested for misuse detection [17-19].

So considering what has been discussed, ANN has been proved to be effective in detecting attacks. RBF as mentioned earlier, was faster with better performance in detecting misuse and abnormal usage. This made us think of a hybrid neural network which has RBF as the base.

Probabilistic Neural Network (PNN) makes training faster. That is the reason we decided to have a combination of these two networks. The number of Gaussians is entered using the Cluster Centers field. It is impossible to suggest an appropriate number of Gaussians, because it is problem dependent. Use of PNN made our simulation faster. In order to evaluate our work we decided to compare our results to two popular method of classification. One of them is SOM which was mentioned earlier. The other one is Support Vector Machine (SVM). One of the approaches which has widely been used in IDS is Support Vector Machine (SVM) [20-22]. It uses a space of linear functions in high dimensional features. It can be effectively used in classification. Our results are compared to SVM and SOM to show its improvement. The paper is organized as follows: In section II, the proposed method is suggested. Simulation can be found in section III and section IV includes conclusion.

II. PROPOSED METHODS

PNN or "Probabilistic Neural Network" is used forkernel analysis. Its a normalized RBF network in which there is a hidden unit centered at every training case. These RBF units are called "kernels" and are usually probability density functions such as the Gaussian. The hidden-to-output weights are usually 1 or 0; for each hidden unit, a weight of 1 is used for the connection going to the output that the case belongs to, while all other connections are given weights of 0. These weights can be adjusted for the prior probabilities of each class. So the only weights that need to be learned are the widths of the RBF units. These widths (often a single width is used) are called "smoothing parameters" or “bandwidths" and are usually chosen by cross-validation or by more esoteric methods that are not well-known in the neural net literature.

Speech’s claimed that a PNN trains 100,000 times faster than back propagation is atbest misleading [23-25]. While they are not iterative in the same sense as backpropagation,kernel methods require estimating the kernel bandwidth and this requires accessing the data many times. Furthermore, computing a single output value with kernel methods requires either accessing the entire training data or clever programming and either way is much slower than computing an output with a feed forward net. There are a variety of methods for training feedforward nets that are much faster than standardbackpropagation. PNN is a universal approximateor for smooth class-conditional densities, so it should be able to solve any smooth classification problem given enough data. The main drawback of PNN is that, like kernel methods in general, it suffers badly from the curse of dimensionality. PNN cannot ignore irrelevant inputs without major modifications to the basic algorithm.

We know that the number of patterns in the training set affects the number of centers (more patterns imply more Gaussians), but this is mediated by the dispersion of the clusters. If the data is very well clustered, then few Gaussians are needed. On the other hand, if the data is scattered, many more Gaussians are required for good performance. For standard RBF's, the supervised segment of the network only needs to produce a linear combination of the output at the unsupervised layer. Therefore no hidden layers is the default. Hidden Layers can be added to make the supervised segment a multi layer perception (MLP)

instead of a simple linear perception. So combination of these layers is supposed to generate a better IDS.

III. SIMULATION

We gathered data from KDD [26]. The 1998 DARPA Intrusion Detection Evaluation Program was prepared and managed by MIT Lincoln Labs. Their purpose was to evaluate research in intrusion detection. A standard set of data which includes a large variety of intrusion simulated in a military network environment was prepared. The 1999 KDD intrusion detection contest uses a version of this dataset. Lincoln Labs set up an environment to gather nine weeks of raw TCP dump data for a local-area network (LAN) simulating a typical U. S. Air Force LAN.

A connection is a sequence of TCP packets starting and ending at some well defined times, between which data flows to and from a source IP address to a target IP address under some well defined protocol. Each connection is labeled as either normal, or as an attack, with exactly one specific attack type. Each connection has 41 features which can be seen in TABLE I.

Table I: FEATURES OF EACH TCP CONNECTION

Feature	Attribute
Duration	Continuous
protocol_type	Symbolic
service	Symbolic
Flag	Symbolic
src_bytes	Continuous
dst_bytes	Continuous
Land	Symbolic
wrong_fragment	Continuous
Urgent	Continuous
Hot	Continuous
num_failed_logins	Continuous
logged_in	Symbolic
num_compromised	Continuous
root_shell	Continuous
su_attempted	Continuous
num_root	Continuous
num_file_creations	Continuous
num_shells	Continuous
num_access_files	Continuous
num_outbound_cmds	Continuous
is_host_login	Symbolic
is_guest_login	Symbolic
Count	Continuous
srv_count	Continuous
serror_rate	Continuous
srv_serror_rate	Continuous
rerror_rate	Continuous
srv_rerror_rate	Continuous
same_srv_rate	Continuous
diff_srv_rate	Continuous
srv_diff_host_rate	Continuous
dst_host_count	Continuous
dst_host_srv_count	Continuous
dst_host_same_srv_rate	Continuous
dst_host_diff_srv_rate	Continuous
dst_host_same_src_port_rate	Continuous
dst_host_srv_diff_host_rate	Continuous
dst_host_serror_rate	Continuous
dst_host_srv_rerror_rate	Continuous
dst_host_rerror_rate	Continuous

We simulated data in NeuroSolution (v6) [27]. In order to evaluate our methods, the following parameters are calculated and the results are shown in TABLE. II.

- True negative rate (TNR): $\frac{TN}{TN+FP}$, also known as specificity.

- True positive rate (TPR): $\frac{TP}{TP+FN}$, also known as detection rate (DR)or sensitivity. In information retrieval, this is called recall.

- False positive rate (FPR): $\frac{FP}{TN+FP}$: 1 _ specificity, also known asfalse alarm rate (FAR).

- False negative rate (FNR): $\frac{FN}{TP+FN}$: 1 _ sensitivity.

TableII : SIMULATION RESULTS FOR RBF/GRNN/PNN

Attack	TPR	TNR	FPR	FNR
Normal	99.6	82.6	17.4	0.4
DoS	95.8	93.54	6.46	4.2
Probe	96.27	97.06	2.94	3.27
R2L	85.7	84.1	15.9	14.3
U2R	96	100	0	4

Our simulation was done in 2 minutes. The mean square error in all our simulations were around 0.000001 to 0.000005 which shows its high accuracy. In order to evaluate our suggested method, we compare our results to SVM and SOM. Self-organizing feature maps (SOFMs) transform the input of arbitrary dimension into a one or two dimensional discrete map subject to a topological (neighborhood preserving) constraint. The feature maps are computed using Kohonen unsupervised learning. The output of the SOFM can be used as input to a supervised classification neural network such as the MLP. This network's key advantage is the clustering produced by the SOFM which reduces the input space into representative features using a self-organizing process. Hence the underlying structure of the input space is kept, while the dimensionality of the space is reduced.

The Support Vector Machine (SVM) is implemented using the kernel Adatron algorithm. The kernel Adatron maps inputs to a high-dimensional feature space, and then optimally separates data into their respective classes by isolating those inputs which fall close to the data boundaries. Therefore, the kernel Adatron is especially effective in separating sets of data which share complex boundaries. SVMs can only be used for classification, not for function approximation. In final we found several result that show in below chart and TABLE V.

We simulated our data with SVM and SOM. The results can be seen in TABLE III and IV.

Table III: SIMULATION RESULTS FOR SUPPORT VECTOR MACHINE

Attack	TPR	TNR	FPR	FNR
Normal	90	38.5	61.5	10
Dos	100	32.3	67.7	0
Probe	83.2	97.06	2.94	16.8
R2L	98.2	77.3	22.7	1. 8
U2R	63.4	95.4	4.6	36.6

Table IV: SIMULATION RESULTS SELF ORGANIZING MAP

Attack	TPR	TNR	FPR	FNR
Normal	89	96	4	11
DoS	51	33	67	49
Probe	100	12	88	0
R2L	50	49	51	50
U2R	70	75	25	30

Table V: Several attack in Net work

Attack	solution	FPR	TRP
Normal	89	90	99.6
DoS	51	100	95.8
Probe	100	83.2	96.27
R2L	50	98.2	85.7
U2R	70	63.4	96

Fig1: **Several attack in Net work**

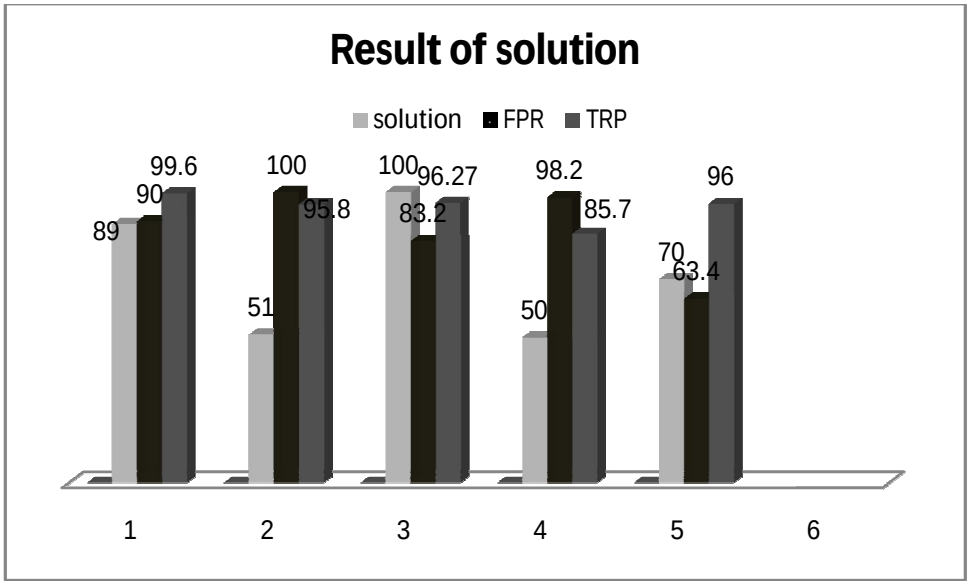

Table V: Result of solution

As it can be seen from tables, DR of RBF/GRNN/PNN is higher than SVM and SOM. RBF/GRNN/PNN's FPR is less than SVM and SOM. This shows that RBF/GRNN/PNN acts more successfully in classification rather than SVM and SOM.

IV. CONCLUSION

The simulation results show that RBF/GRNN/PNN has better performance comparing to support vector machine and self organizing map. This is proved by higher DR and lower FPR. This illustrates that RBF/GRNN/PNN acts more successfully in classification. This solution is comfortable for use in some network such as IMS and NGN network. IMS network have been under several attack same as DOS offline. this solution is very useful for IMS network protected.This solution have been checked in Neural network and have been checked for several attack in NGN network.

REFERENCES

[1] S. X. Wu, W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," *Applied Soft computing*, vol. 10, pp. 1-35, 2010.

[2] J. Ryan, M.J. Lin, and R. Miikkulainen, "Intrusion detection with neural networks," *Advances in Neural Information Processing Systems*, vol. 10, pp. 943–949, 1998.

[3] K. Tan, "The application of neural networks to unix computer security,"*Proceedings of IEEE International Conference on Neural Networks, vol. 1, Perth,WA, Australia, November/December 1995*, IEEE Press, pp. 476–481, 1995.

[4] A.K. Ghosh, A. Schwartzbard, "A study in using neural networks for anomaly andmisuse detection," *Proceedings of the 8th USENIX Security Symposium*, vol. 8,Washington, DC, USA, 23–36 August, pp. 141–152, 1999.

[5] A.K. Ghosh, J. Wanken, and F. Charron, "Detecting anomalous and unknown intrusionsagainst programs,"*Proceedings of the 14th Annual Computer Security Applications Conference (ACSAC'98)*, Phoenix, AZ, USA, 7–11 December 1998, IEEEComputer Society, pp. 259–267,1998.

[6] J. Jiang, C. Zhang, and M. Kame, "RBF-based real-time hierarchical intrusion detectionsystems,"*Proceedings of the International Joint Conference on Neural Networks (IJCNN'03)*, vol. 2, Portland, OR, USA, 20–24 July, IEEE Press, pp.1512–1516, 2003.

[7] E. Leon, O. Nasraoui, and J. Gomez, "Anomaly detection based on unsupervised nicheclustering with application to network intrusion detection,"*Proceedings of the IEEE Congress on*

EvolutionaryComputation (CEC'04), vol. 1, Portland, OR,USA, 19–23 June 2004, IEEE Press, pp. 502–508, 2004.

[8] A. Hofmann, C. Schmitz,and B. Sick, "Rule extraction from neural networks forintrusion detection in computer networks,"*IEEE International Conference on Systems*, Man and Cybernetics, vol. 2, 5–8 October 2003, IEEE Press, pp.1259–1265, 2003.

[9] J. Jiang, C. Zhang, and M. Kame, "RBF-based real-time ierarchical intrusion detectionsystems,"*Proceedings of the International Joint Conference on Neural Networks (IJCNN'03)*, vol. 2, Portland, OR, USA, 20–24 July, IEEE Press, pp.1512–1516, 2003.

[10] Z. Liu, G. Florez, and S.M. Bridges, "A comparison of input representations in neuralnetworks: a case study in intrusion detection,"*Proceedings of the International Joint Conference on Neural Networks (IJCNN'02)*, vol. 2, Honolulu, HI, USA,12–17 May 2002, IEEE Press, pp. 1708–1713, 2002.

[11] Z. Zhang, J. Li, C. Manikopoulos, J. Jorgenson, and J. Ucles, "IDE: a hierarchicalnetwork intrusion detection system using statistical preprocessing and neuralnetwork classification,"*Proceedings of the 2001 IEEE Workshop Information Assurance and Security*, West Point, NY, USA, IEEE Press, pp. 85–90, 2001.

[12] Y. Yu, F. Gao, and Y. Ge, "Hybrid BP/CNN neural network for intrusion detection,"*Proceedings of the 3rd International Conference on Information security*, vol. 85 of ACM International Conference Proceeding Series, pp. 226–228, 2004.

[13] T. Kohonen, "Self-organizing Maps," vol. 30 of Springer Series in InformationSciences, Springer, No. 3, Berlin, 2001.

[14] K. Fox, R. Henning, and J. Reed, "A neural network approach toward intrusion detection,"*Proceedings of the 13th National Computer Security Conference*, vol. 1,Washington, DC, USA, 1–4 October 1990, pp. 124–134,1990.

[15] A.J. Hoglund, K. Hatonen, and A.S. Sorvari, "A computer host-based user anomalydetction system using the self-organizing map," *Proceedings of the IEEEINNS-ENNS International Joint Conference on Neural Networks (IJCNN'00)*, vol.5, Como, Italy, 24–27 July 2000, IEEE Press, pp. 411–416, 2000.

[16] W. Wang, X. Guan, X. Zhang, and L. Yang, "Profiling program behavior for anomalyintrusion detection based on the transition and frequency property of computeraudit data,"*Computers & Security*,vol. 25, No. 7, pp. 539–550, 2006.

[17] J. Cannady, J. Mahaffey, "The application of artificial neural networks to misusedetection: initial results," *Proceedings of the 1st International Workshop on Recent Advances in Intrusion Detection (RAID 98)*, Louvain-la-Neuve, Belgium,14-16 September 1998, 1998.

[18] A. Bivens, C. Palagiri, R. Smith, B. Szymanski, and M. Embrechts, "Networkbasedintrusion detection using neural networks,"*Intelligent Engineering Systems through Artificial Neural Networks*,vol. 12, No. 1, pp. 579–584, 2002.

[19] C. Jirapummin, N. Wattanapongsakorn, and P. Kanthamanon, "Hybrid neural networksfor intrusion detection system,"*The 2002 International Technical Conference on Circuits/Systems, Computers and Communications (ITCCSCC'02)*, vol. 7, Phuket, Thailand, 2002, pp. 928–931, 2002.

[20] Nello Cristianini and John Shawe-Taylor, "AnIntroducton to Support Vector Machines and other kernelbased learning methods", *Tenth Reprint, Cambridge, University PressHand*, 2006.

[21] Pai-Hasuen Chen, Chih-jen lin , and Bernhard, "Atutorial on support Vector Machine", *Department ofComputer Science & Information Engineering*, NationalTaiwan University.

[22] S Mukkamala, G Janoski, A Sang "Intrusion Detectingusing Neural Network and Support Vector Machine",*Proceeding of IEEE International Joint Conference inNeural Network*, pp 1702-1707, 2002.

[23] D.J., "Kernel Discriminant Analysis,"*Research Studies Press*,1982.

[24] Lowe, D.G., "Similarity metric learning for a variable-kernel classifier," *Neural Computation*, vol. 7, pp. 72-85, 1995.

[25] McLachlan, G.J.,"Discriminant Analysis and Statistical Pattern Recognition,"*Wiley*, 1992.

[26] The KDD99 Dataset. Retrieved January 26, 2008, from http://kdd.ics.uci.edu/databases/kddcup99/task.html.

[27] NeuroSolution, version 6, by Curt Lefebvre and Jose Principe, NeuroDimension Inc, Online:www.ns.com.