An Ensemble Classification Approach for Intrusion Detection

Riyad.A.M
Research scholar
Bharathiar University, Coimbatore,
India

M.S Irfan Ahmed, Ph.D
Director, MCA
Hindustan College of Engineering
and Technology, Coimbatore,
India

ABSTRACT

Increased cyber attacks in various forms compel everyone to implement effective intrusion detection systems for protecting their information wealth. From last two decades, there has been extensive research going on in intrusion detection system development using various techniques. But, designing detection systems producing maximum accuracy with minimum false positive is yet a challenging task for the research community. Ensemble method is one of the major developments in the field of machine learning. In this research work, new ensemble classification method is proposed from different classifiers. Support vector machine techniques, artificial neural network and random forest are used for classification. Ensemble model is formed for producing better result. The model shows promising result for all classes of attacks.

General Terms

Computer Security, Intrusion Detection Systems, Ensemble classification.

Keywords

Intrusion detection, classification, ensemble, particle swam optimization, support vector machine, SVM, ANN, RS.

1. INTRODUCTION

One of the biggest threats faced by the modern era of computing is obviously the attacks through the networks. This leads to undependable and inconsistent states of systems causing far reaching consequences on various domains, making the whole networking sphere worthless. Intrusions are different types of attacks on targeted devices in order to affect their integrity, confidentiality and availability. Intrusion detection is the act of detecting the intrusions through various techniques. Intrusions can be found by tracing the anomalous network activities. An intrusion detection system (IDS) monitors network or system activities for malicious actions and produces reports to an authority. If finding the intrusion is challenging, avoiding a normal activity misjudged as intrusion will be more challenging.

Intrusion detection is the process of identifying the intrusions through various techniques. Intrusions can be found by tracing the anomalous network activities or by verifying the data in the host machine. Different methods are devised to identify intrusions. We can broadly classify them into misuse detection and anomaly detection. Misuse detection techniques trace the abnormalities through matching anomalous signatures of previous attacks with current patterns. This is more similar to antivirus logic. On the other hand, Anomaly detection techniques catch hold on all activities other than normal ones [1]. Hence, here the normal profiles are well identified first to observe deviations of current activities if any. It is the responsibility of this technique to make sure that the deviations identified is well enough to label as 'intrusion'. The quality of the approaches towards detection of anomalous activities can be measured by observing the false positives and false negatives. False positive is the scenario where an event is incorrectly identified by the system as being an intrusion when none has occurred. False negative is the situation where no intrusion has been identified by the system when one has in fact occurred.

The objective of this paper is to maximize the accuracy of the detection system. For that purpose, an ensemble approach is introduced that can outperform individual approaches in terms of accuracy in detection of intrusions. First, relevant features are selected from the traffic patterns and then classifiers are constructed from SVM, ANN and RF methods for engineering the proposed ensemble classification model.

The rest of the paper is organized as follows: Section 2 discusses the related works in the area. Section 3 discusses the KDD cup dataset and its features. Section 4 discusses the techniques used in this paper such as SVM, ANN and RF. Section 5 discusses the ensemble approach and the system architecture. Section 6 describes the experimental setup and the paper is concluded in section 7. Section 8 lists the references.

2. RELATED WORKS

As interest in intrusions increases, so does the intrusion detection. James P Anderson in his technical report in 1980 introduced the idea of intrusion detection systems [2]. He stated important aspects of host based intrusion detection such as audit trails and evaluation of log files. In 1987, Dr. Dorothy Denning designed a model which helped as a guideline for development of current commercial IDS [3]. Stephen E. Smaha modeled individual users by assigning profiles and updating them [4]. It used a statistical anomaly detection algorithm. Artificial neural network was efficiently utilized by Hansen and Salamon [5] which increased the classification accuracy. An approach of user behavior modeling which takes advantages of properties of neural algorithms was proposed by H Debar et al. [6].

Weak learning algorithms can be combined to obtain high accuracy. The strength of each algorithm in ensemble can be utilized for obtaining a robust classifier. Another advantage of ensemble is that each sub problems can be handled by a particular or a group of algorithms suitable to the scenario. Linear genetic programming was utilized by Mukkamal et al. [7] for modeling intrusion detection system with support of ensemble technique. Cherbrolu et al. [8] explains the advantages of combining redundant and complementary classifiers for increasing accuracy. Performance evaluation of linear genetic algorithm, multi expression programming and gene expression programming was conducted by Abraham et al. [9][10]. Zainal et al. compared the results of different machine learning algorithms such as linear genetic programming, adaptive neural fuzzy interface system and random forest [11]. Bahri et al. introduced a novel method based on boosting technique which was an adaptation of Adaboost [12]. Recently, there are many papers published regarding ensemble approaches [13][14]. Performance evaluation of a distributed system of ensemble known as GEdIDS was done by Folino et al.[15][16].

3. KDD CUP DATA SET

KDD 99 is a readymade dataset which is widely used by the research community for analyzing abnormal patterns and comparing the results inorder to know the performance of algorithms performed on this data set. It is a bench mark data set used by various intrusion detection systems. The data set contains 41 features.

The data set is generated by creating an environment with three target machines. Three systems were used for spoofing and a sniffer for reading the network traffic [17].

3.1 Attack categories

The four categories of attacks are,

1) Denial of Service Attack (DoS): Denial of Service Attack (DoS): This is the attack accomplished by making the computing resources busy, so that genuine requests are denied.

2) User to Root Attack (U2R): Here, user accounts are attained by the attacker through illegal means and this is used for accessing the privileges of root account.

3) Remote to Local Attack (R2L): Here, the attacker utilizes the vulnerabilities of a system for gaining illegal access.

4) Probing: Here the attacker illegally monitors the network devices and collect information in the intension of breaching its security.

KDD attacks are detailed in table1. Various attacks coming under four categories are listed in table2. Table 3 lists 41 features of the data set.

Table 1

Attack	Number of samples
normal	97277
smurf	280790
neptune	107201
back	2203
teardrop	979
pod	264
land	21
satan	1589
ipsweep	1247
portsweep	1040
nmap	231
warezclient	1020
guess_passwd	53
warezmaster	20
Imap	12
ftp_write	8
multihop	7
Phf	4
Spy	2
buffer_overflow	30
rootkit	10
loadmodule	9
Pearl	3

Table 1 depicts KDD attacks and **Table 2** describes various types of attacks under four major categories

Table 2

Denial of Service (DOS)	Back, land, Neptune, pod, smurf, teardrop
User to Root	Buffer_overflow, loadmodule, perl, rootkit
Remote to Local	ftp_write, guess_passwd, imap, multihop, phf,spy, warezclient, warezmaster
Probe	Satan, ipsweep, nmap, portsweep

Table 3 below lists the 41 features and descriptions (type C is continuous, while D is discrete)

Table 3

#	Feature name	Description	Type
1	duration	Length (# of seconds) of the connection	C
2	protocol type	Type of the protocol, e.g. tcp, udp, etc.	D
3	service	Network service on the destination, e.g., http, telnet, etc.	D
4	flag	Normal or error status of the connection	D
5	src_bytes	#ofdatabytesfromsourcetodestination	C
6	dst_bytes	# of data bytes from destination to source	C
7	land	1 if connection is from/to the same host/port; 0 otherwise	D
8	wrong_fragme nt	# of “wrong” fragments	C
9	urgent	# of urgent packets	C

10	hot	# of "hot" indicators	C
11	num_failed_log ins	# of failed login attempts	C
12	logged in	1 if successfully logged in; 0 otherwise	D
13	num_comprom ised	# of compromised conditions	C
14	root_shell	1 if root shell is obtained; 0 otherwise	D
15	su_attempted	1 if "su root" command attempted; 0 otherwise	D
16	num_root	# of "root" accesses	C
17	num_file_creations	# of file creation operations	C
18	num_shells	# of shell prompts	C
19	num_access_files	# of operations on access control files	C
20	num_outbound_cmds	# of outbound commands in an ftp session	C
21	is_host_login	1 if the login belongs to the "hot" list; 0 otherwise	D
22	is_guest_login	1 if the login is a "guest' login; 0 otherwise	D
23	count	# connections to the same host as the current one during past two seconds	C
24	srv_count	# of connections to the same service as the current connection in the past two seconds	C
25	serror_rate	% of connections that have "SYN" errors	C
26	srv_serror_rate	% of connections that have "SYN" errors	C
27	rerror_rate	% of connections that have "REJ" errors	C
28	srv_rerror_rate	% of connections that have "REJ" errors	C
29	same_srv_rate	% of connections to the same service	C
30	diff_srv_rate	% of connections to different services	C
31	srv_diff_host_rate	% of connections to different hosts	C
32	dst_host_count		C
33	dst_host_srv_count		C
34	dst_host_same_srv_rate		C
35	dst_host_diff_srv_rate		C
36	dst_host_same_src_port _rate		C
37	dst_host_srv_diff_host_ rate		C
38	dst_host_serror_rate		C
39	dst_host_srv_serror_rate		C
40	dst_host_rerror_rate		C
41	dst_host_srv_rerror_rate		C

4. APPROACHES USING SVM, RF AND ANN

4.1 Support vector machine

Support vector machine is a powerful algorithm in machine learning for the pattern recognition. It is based on supervised learning technique. The algorithm was prepared by Vapnik [18][19]. Non linear input vector is mapped into high dimensional feature space. This robust algorithm maximizes the classification by sub dividing feature space into sub spaces.

Here, a model is build which classifies new data. Non linear and linear classification can be done by SVM. While non linear, the solution is formed by extending the original set of variable x in a high dimensional feature space with map Φ. If input vector x$\in R^d$is transformed to feature vector Φ(x) by a map Φ: $R^d \rightarrow H$, then a function can be found, K $(R', R') \rightarrow R$ that satisfies condition K(x_i, x_j)= Φ(x_i).Φ(x_j) and problem leads to the following quadratic optimization problem,

Minimize $\sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j \, y_i y_j (x_i x_j)$ [20][21][22]

Subject to $\sum_{i=1}^{k} y_i \alpha_i = \forall i: 0 \leq \alpha_i \leq C$

The generalization highly depends on the geometrical nature of data rather than dimensionality.

4.2 Random Forest

Random forest is combination of tree predictors. Random vector is sampled for each tree. The tree depends on these values. Predictors are randomly chosen for generating trees.

The RF algorithm constructs the tree with different bootstrap samples. After the construction of the tree, data can be given as input to the tree for classification. Each tree gives a vote regarding the classification of the data. Decision of the class is done by considering the majority of votes. The algorithm is illustrated below [23].

1. Construct bootstrap sample S_i from the dataset D, where |S_i|=|D| and random examples are chosen with replacement from D.

2. Construct a tree T using S_i using standard decision tree algorithm with following modifications.
   a. At each, node restricts the candidate attribute to a randomly selected subset $(x_1, x_2, x_3, \ldots, x_k)$, where k = number of features.
   b. Do not prune the tree.
3. Repeat steps 1 and 2 for i =1, … , number of trees, creating a forest of trees T_i derived from different bootstrap samples.
4. When classifying a sample x, aggregate the vote for all trees Ti in the forest. If T_i (x) is the class of x as determined by the tree T_i. Then the predicted class of x is the class with the majority of votes.

4.3 Artificial Neural Network

Artificial neural network is designed from the inspiration gained from human neuron system. The processing elements are called neurons. It process information. The power of neural network increases when combining neurons into multilayer set-up.

Here, multilayer neural network use back propagation. The algorithm is a supervised one. There will be forward and backward pass. During forward pass, forward neural network computation is done and it is propagated through the network.[24][[25]. Here, the synaptic weights are fixed. During the backward pass, error correction rule is used for adjusting the synaptic weight.

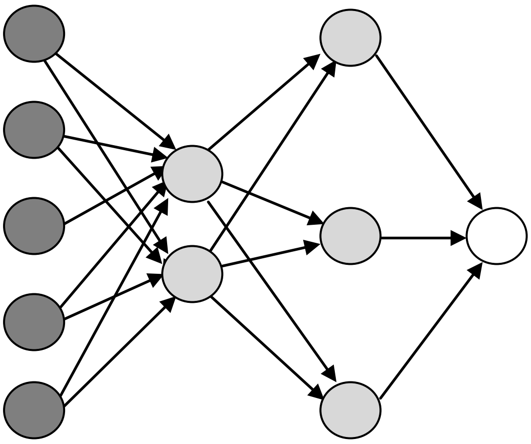

Fig 1 ANN with two hidden layers

The neuron activation functions are given below [24].

Sigmoidal : f(x) = $\frac{1}{1+\exp(-ax)}$, a > 0

Tansig: f(x) = a tanh(bx) , a & b > 0

In short, a back propagation network learns through examples. It can be trained by adjusting weights and finally it can give the required output for the given input.

5. THE ENSEMBLE METHOD

An ensemble model utilizes some base models to classify the data. There are different techniques established for learning ensemble models and using them in combination. Bagging, boosting and stacking with their variants are efficient among them. These models can increase the accuracy of prediction over a single model.

An ensemble of classifiers is a set of classifiers where individual decisions are combined to classify new examples [26]. In our approach, three base models of classification are used. The model learns the dataset and individually classifies the data .In the ensemble approach, the final classification is decided as follows.

1. Each classification from the base algorithms is given a weight 0 to 1 depending on their accuracy.
2. When classifiers agree, the decision is made according to their classification.
3. If the classifiers disagree to each other, the classifier with highest weight is considered.

The architecture of model is shown below.

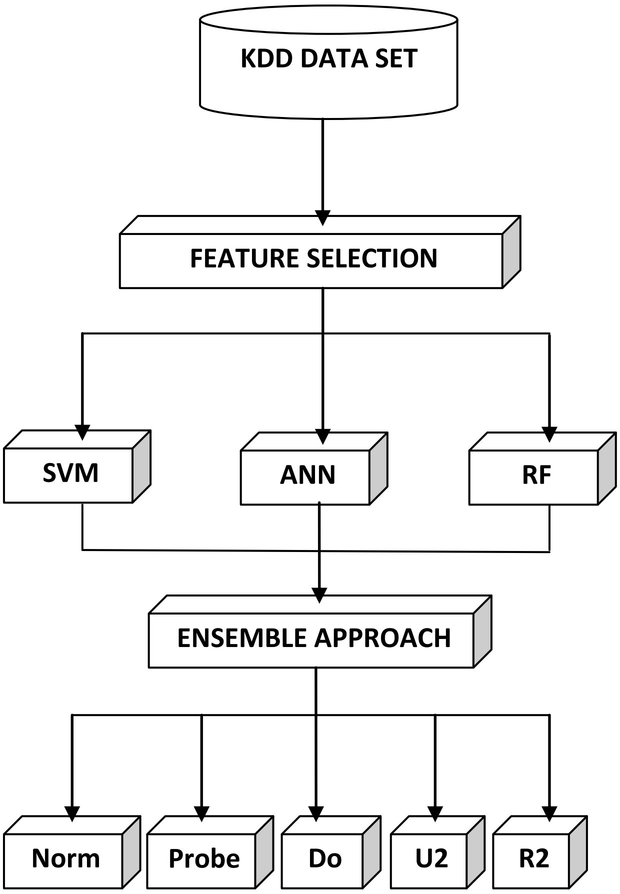

Fig 2 showing the architecture of the model.

6. EXPERIMENTAL SETUP

The KDD cup data set is used with its 41 features. Classification accuracy was evaluated using 10-fold cross validation. At first, the base classifiers such as support vector machine, artificial neural network and random forest are considered individually. After that, the ensemble of SVM, RF and ANN are modeled.

All experiments are conducted in WEKA (Waikato Environment for Knowledge Analysis) 3.5.7 designed by machine learning group at University of Waikato. Random forest in this environment is a variant of REPTree algorithm and is not so robust. Hence, slight modifications were made. Back propagation version of ANN is implemented in this work. SMO algorithm is used for training of SVM.

Table 4 below showing the performance comparison.

Table 4

	SVM accuracy	ANN accuracy	RF accuracy	Ensemble accuracy
Normal	99.03	82.21	91.16	99.61
Probe	86.06	99.81	93.76	99.83
DoS	89.36	97.91	88.45	97.99
U2R	99.72	66.32	97.13	98.87
R2L	65.77	96.61	94.87	96.89

It shows the classification accuracy of individual approaches and the ensemble model. The ensemble model outperforms the individual approaches. This is perhaps due to the complementary role from each of the base classifiers implemented.

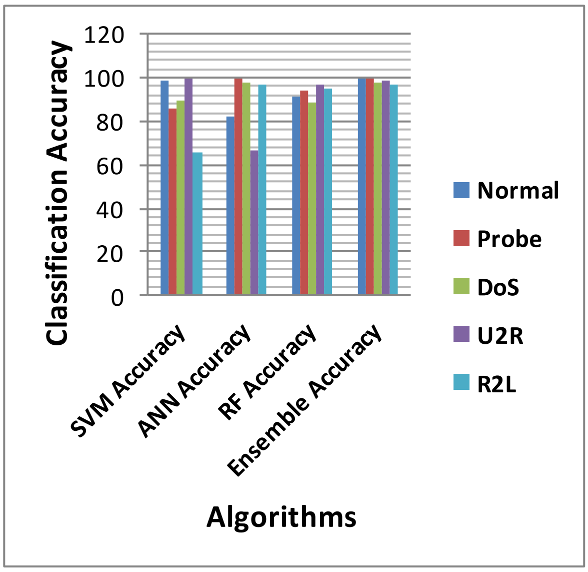

Fig 3 showing the performance of each technique.

7. CONCLUSION

KDD cup data set is used here for evaluating individual performance of the algorithm as well as the ensemble model. We used SVM, ANN and RF as individual detection models. The performance comparison states that the ensemble model shows better performance than the individual algorithms in detecting attacks. Architecture of base classifiers and their ensemble is proposed. We have not given much importance for reducing false positives. Also, we can increase the performance of the system with reduced features. Our future research would be directed towards this.

8. REFERENCES

[1] Gogoi P, Borah B, Bhattacharyya D. "Anomaly detection analysis of intrusion data using supervised & unsupervised approach." Journal of Convergence Information Technology 2010.

[2] J. P. Anderson. "Computer security threat monitoring and surveillance". Technical report, James P. Anderson Company, Fort Washington, Pennsylvania, April 1980.

[3] Dorothy E. Denning. An intrusion-detection model. IEEE Trans. Software Eng., 1987.

[4] Stephen E. Smaha, "Haystack: An intrusion detection system." In Proceedings of the Fourth Aerospace Computer Security Applications Conference, December 1988.

[5] Lars Kai Hansen and Peter Salamon. "Neural Network Ensembles." IEEE Transactions on Pattern Analysis and Machine Intelligence, October 1990.

[6] H. Debar, M. Becker and D. Siboni, "A Neural Network Component for an IntrusionDetection System", In Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, CA, May 1992.

[7] S. Mukkamala, A.H. Sung and A. Abraham, "Modeling Intrusion Detection Systems Using Linear Genetic Programming Approach." LNCS 3029, Springer Hiedelberg, 2004.

[8] S. Chebrolu, , A. Abraham, and J.P. Thomas, "Feature Deduction and Ensemble Design of Intrusion Detection Systems." International Journal of Computers and Security, Vol 24, Issue 4, June 2005.

[9] Ajith Abraham and Crina Grosan. "Evolving Intrusion Detection Systems", volume 13 of Studies in Computational Intelligence, Springer-Verlag, Berlin, Heidelberg, 2006.

[10] Ajith Abraham, Crina Grosan, and Carlos Martin-vide. "Evolutionary Design of Intrusion Detection Programs." International Journal of Network Security, November 2006.

[11] Anazida Zainal, Mohd Aizaini Maarof, Siti Mariyam Shamsuddin, and Ajith Abraham. "Ensemble of One-Class Classifiers for Network Intrusion Detection System." In Proccedings of the 4th International Conference on Information Assurance and Security (IAS), IAS '08, IEEE Computer Society. Napoli, Italy, September 2008.

[12] Emna Bahri, Nouria Harbi, and Hoa Nguyen Huu. "Approach Based Ensemble Methods for Better and Faster Intrusion Detection." In Proceedings of the 4th International Conference on Computational Intelligence in Security for Information Systems, Lecture Notes in Computer Science, Torremolinos-Malaga, Spain, June 2011. Springer.

[13] Silvia Gonz´alez, Javier Sedano, Alvaro Herrero, Bruno Baruque, and Emilio Corchado. "Testing ensembles for intrusion detection: On the identification of mutated network scans." In Proceedings of the 4th international conference on Computational intelligence in security for information systems, CISIS'11, Torremolinos-Malaga, Spain, June 2011. Springer-Verlag.

[14] Peng Zhang, Xingquan Zhu, Yong Shi, Li Guo, and Xindong Wu. “Robust ensemble learning for mining noisy data streams.” Decision Support Systems, January 2011.

[15] Gianluigi Folino, Clara Pizzuti, and Giandomenico Spezzano. “GP Ensemble for Distributed Intrusion Detection Systems.” In Proceedings of the 3rd International Conference on Advances in Pattern Recognition (ICAPR), Bath, UK, August 2005.

[16] Gianluigi Folino, Clara Pizzuti, and Giandomenico Spezzano. “An ensemble-based evolutionary framework for coping with distributed intrusion detection.” Genetic Programming and Evolvable Machines, June 2010.

[17] H.G. Kayacik, A.N. Zincir-Heywood, and M.I. Heywood, “Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets”, In Proceedings of the 3rd Annual Conference on Privacy, Security and Trust (PST-2005), Oct., 2005

[18] V. Vapnik,. “Statistical Learning Theory.” Wiley, New York, 1998

[19] Vapnik V. “The nature of statistical learning theory.” New York: Springer; 1995.

[20] Kuan-Ming Lin and Chih-Jen Lin, “A Study on Reduced Support Vector Machines”, IEEE Transactions On Neural Networks, VOL. 14, NO. 6, NOVEMBER 2003.

[21] R.Debnath, H.Takahashi, “SVM Training: Second-Order Cone Programming versus Quadratic programming”, 2006 IEEE International Joint Conference on Neural Networks, Canada, July 16-21, 2006.

[22] Arvind Mewada, PraffulGedam, Shamaila khan,M.Udayapal reddy, ” Network Intrusion Detection Using

Multiclass Support Vector Machine”, Special Issue of IJCCT Vol.1 Issue 2, 3, 4; 2010 for International Conference [ACCTA-2010], 3-5 August 2010

[23] T.M. Khoshgoftaar, M. Golawala and J. Van Hulse, “An Empirical Study of Learning from Imbalanced Data Using Random Forest." In Proceedings of the 19th. IEEE Conference on Tools with Artificial Intelligence, 2007.

[24] Vu N. P. Dao 1, Rao Vemuri, “A Performance Comparison of Different Back Propagation Neural Networks Methods in Computer Network Intrusion Detection”, Differential Equations and Dynamical System 2002

[25] J. Principe, N. Euliano, W. Lefebvre, “Neural and Adaptive System – Fundamentals Through Simulations”, Wiley, 2000.

[26] P.Kang, and S.Cho, “EUS SVMs: Ensemble of Under Sampled SVMs for Data Imbalance Problems.” ICONIP 2006.