ISSN No. 0976-5697

Volume 7, No. 4, July-August 2016

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

FUZZY BASED SUPPORT VECTOR MACHINE CLASSIFIER WITH WIENER FILTER (FSVM – WF) FOR INTRUSION DETECTION SYSTEM

R. Karthik
Assistant Professor,Department of IT
Kongunadu Arts and Science College (Autonomous)
Coimbatore, India.

Dr.S.Veni
Professor & Head
Department of Computer Science
Karpagam Academy of Higher Education,Coimbatore, India.

Dr.B.L.Shivakumar
Principal
Sri Ramakrishna Polytechnic College, Coimbatore,India.

Abstract : This research work aims in design and development of fuzzy based support vector machine classifier with wiener filter (FSVM – WF) for intrusion detection system. The proposed research work performs pre-processing using wiener filter. Then fuzzy logic system is added up with support vector machine for performing the classification task. KDD cup'99 dataset that contains four major types of attacks in the network is chosen for performing FSVM-WF classification. Performance metrics detection rate and false alarm rate are chosen. Simulation results shows that the proposed FSVM-WF classifier outperforms in terms of improved detection rate and reduced false alarm rate.

Keywords: Wiener Filter, Intrusion Detection System, Fuzzy Logic System, KDD cup'99, FSVM-WF.

I. INTRODUCTION

Intrusion Detection Systems (IDSs) is an advancing innovation for ensuring computer networks. Case in point, in prior day's denial-of-service (DoS) attack can't bring about genuine fiascos, however today, fruitful DoS attacks can bring about awesome money related misfortune to associations. The objective of interruption detection frameworks is to recognize peculiar or abuse conduct of framework and tell to network directors about the exercises. Numerous interruption detection devices have security shortcomings, for example, neglecting to encrypt the log documents, overlooking access control, and neglecting to perform trustworthiness checks, and so on. An IDS is more secure than other security instruments, for example, firewalls [1]. Prior research framework in light of two noteworthy ideas known as anomaly detection and signature detection in view of anomalous conduct of the framework [2]. At first IDS comprises of accumulation of audit data from the watched framework. At that point this data is either preprocessed or specifically connected to the indicator to generate an alarm. The fundamental point of IDS is to expand detection rate and to decrease false alarm rate in recognizing attacks. As of late, the researcher for the most part centered on anomaly detection in view of proposed procedures, for example, data mining, neural system, etc.

The Intrusion detection models can be broadly classified into two namely misuse-based and anomaly-based [3, 4]. A misuse-based intrusion detection system (which is also called as signature-based or pattern-based) will identify attacks that are already known. Misuse based IDS will detect the happened attacks based on information stored in a database. The second type of IDS named Anomaly-based IDS is capable enough to identify / detect both known and unknown intrusions, detecting deviations from normal connections. There exist certain challenges in existing anomaly intrusion detection systems. The challenges are low detection rates and high false alarm rates. Low detection rate can more possibly fail to spot in detecting serious attacks and the high 'false alarm' rates will recognize a normal connection may be falsely classified as an attack. In general, attacks can be divided into four categories [5]:

A. ***Denial of Service (DoS)***

This type of attack is common at the scenario when an attacker intends to deny / restrict authorized users from using a service, computer or resource. Some of the examples of DoS are SYN Flood, Ping of Death, Back, Smurf, Land, Apache2 and Teardrop.

B. ***Remote to User (R2L)***

This type of attack probably happens when an attacker indents to obtain admittance to the victim host. Some of the examples are Sendmail, dictionary, Named, Guest, Imap, Ftp_write.

C. ***User to Root (U2R)***

This type of attack occurs when an attacker gains local access to the victim machine along with intends to obtain super-user / administrator privileges. Examples are Perl, Xterm, Loadmodule, Eject, and Fdformat.

D. ***Probing (Probe)***

This attack is quite common when an attacker intends to take over information through access privileges on the target host. Examples are Saint, Nmap, Mscan, Satan, and Ipsweep.

II. LITERATURE REVIEW

Intrusion detection systems (IDS) are security management systems that are deployed in a network in order to spot out irregular / strange activities and incomplete signatures within computers or networks [6]. The number of methods and frameworks has been proposed and many systems have been built to detect intrusions. The various existing techniques and frameworks are discussed as follows.

Lee et al. have proposed data mining approaches for detecting intrusions in [7], [8], and [9]. Data mining

approaches for intrusion detection makes use of designing classifiers. These are done by setting up pertinent patterns of program along with users' code and conduct. Association rules [10] and frequent episodes are also implemented in the literatures in order to study the record patterns. These record patterns are studied for portraying users' behavior. Association rules and frequent episodes are capable enough to handle symbolic data. These features are best described in the form of packet and connection details. On the other hand, mining of features is only limited to entry level of the packet and needs the number of records to be large and sparsely populated; if not, they be inclined to fabricate a large number of rules that increase the complexity of the system [11].

There exists data clustering methods in the literatures namely k-means and fuzzy c-means algorithms that are also implemented extensively for developing intrusion detection [12], [13]. The major pitfall the above said clustering technique is most of them are based on calculating numeric distance between the observations. Observations along with symbolic features are not capable enough to be ease for clustering task which leads to inaccuracy. In addition, the clustering methods consider the features independently and are unable to capture the relationship between different features of a single record, which further degrades attack detection accuracy.

Naive Bayes classifiers have also been used for intrusion detection [14]. However, they make strict independence assumption between the features in an observation resulting in lower attack detection accuracy when the features are correlated. Bayesian network can also be used for intrusion detection [15]. On the other hand, they are inclined to be attack specific and put up a decision network depending on unusual uniqueness of each and every attack. Hence, the size of a Bayesian network will grow quicker as the number of features and the type of attacks modeled by a Bayesian network increases. To detect anomalous traces of system calls in privileged processes [16], hidden Markov models (HMMs) have been applied in [17], [18], and [19]. Nevertheless, modeling only the system calls will never offer precise / correct classification as in such cases various connection level features are ignored. Further, HMMs are generative systems and fail to model long-range dependencies between the observations [20]. Various research works have been carried out for intrusion detection system using support vector machines [21] – [25].

III. PROPOSED WORK

This research work aims in design and development of fuzzy based support vector machine classifier with wiener filter (FSVM – WF) for intrusion detection system. The proposed research work performs preprocessing using wiener filter as discussed in section 3.1. Then fuzzy logic system is added up with support vector machine (as discussed in section 3.2.) for performing the classification task.

A. Wiener Filter

Wiener filter is used for estimates linear desired signal from other related noisy signal. It plays a vital role in linear estimation, signal restoration and system identification. The main idea of using wiener filter is to calculate minimum mean square error based on the average square distance between the normal filter output and desired output. The signal component $X_i(n)$ be a N-dimensional dataset based on linear combination of normal data $S_i(n)$ and anomaly data $A_i(n)$, expressed mathematically by

$$X_i(n) = S_i(n) + A_i(n) \qquad (1)$$

The wiener filter is mainly used to filter out normal and abnormal data and used to calculate minimum mean square error $e_{min}(n)$ based on normal output data $y_i(n)$ and desired output r (n). The error value is calculated using

$$e_{\min}(n) = r\ (n) - y_i(n) \qquad (2)$$

Rearranging equation (2) we get

$$r(n) = y_i(n) + e_{\min}(n) \qquad (3)$$

Let E_{MSE} denotes the Minimum Mean Square Error, defined by:

$$E_{MSE} = E[|\, e_{\min}(n)\,|^2] \qquad (4)$$

Hence, by evaluating the MSE on both sides of (3), and applying it to the principle of orthogonality we get:

$$\sigma_r^2 = \sigma_{y_i}^2 + E_{MSE} \qquad (5)$$

Where σ_r^2 , $\sigma_{y_i}^2$ is the variance of the desired response and estimated output; on assuming the random variable to zero, the equation (5) is

$$E_{MSE} = \sigma_r^2 - \sigma_{y_i}^2 \qquad (6)$$

The mean square value will lies between zero and one, on dividing Eq (6) be divide by σ_r^2 obtaining

$$\frac{E_{MSE}}{\sigma_r^2} = 1 - \frac{\sigma_{y_i}^2}{\sigma_r^2} \qquad (7)$$

Clearly, this is possible because σ_r^2 is never zero, except when r(n) is zero for all n. Let

$$\kappa = \frac{E_{MSE}}{\sigma_r^2} \qquad (8)$$

Where *k* be normalized mean-squared error, in terms of rewriting (8) in the form:

$$\kappa = 1 - \frac{\sigma_{y_i}^2}{\sigma_r^2} \qquad (9)$$

Where *k* be positive between the $0 \le \kappa \le 1$

To find the normal and abnormal states of the network, the autocorrelation matrix R admits the eigen-decomposition. For instance, the eigenvector places a boundary between normal data traffic and abnormal traffic based on minimum and maximum values. The values beyond maximum may leads to abnormality of data flow. The design of a Wiener filter needs *a priori* knowledge about the statistics of the data to be processed. The filter is optimum only when the statistical features of the input data relate to *a priori* information about the filter. When this information is not known completely, it may not be possible to design the Wiener filter to be optimum.

B. Fuzzy based SVM

Support Vector Machines (SVM) is specifically used to solve a binary classification problem in a supervised manner and the learning problem is formulated as a quadratic optimization problem where the error surface is free of any

local minimum and has global optimum. SVM works on the basis of the principle of structural risk minimization. The main advantage of SVM can train with a large number of patterns. Fuzzy systems based on fuzzy if-rules have been successfully used in many applications areas.

Fuzzy support vector machine proposed in this research work aim to lessen the training time of the classifier. It also elevates the efficiency of the classifier. The fraction of IDS over the total number of them that predicts a given event will determine whether such event is predicted or not. The final step of pre-processing is scaling the training data. During training the data normalizing all features is carried out which will result in null mean (also called as zero mean) and 1 as the standard deviation off. During SVM calculation of SVM, the above said feature of the fuzzy support vector machine eliminates numerical instabilities. We then used the same scaling of the training data on the test set. Once when the salient features are extracted (in terms of the values of the parameters θj), the parsimonious fuzzy rules are applied based on the support vectors $S\{x_s^l\}$,which lies as l=1 and N_s discovered by the SVM.

The training process is performed as follows:

1) Each support vector corresponds to a fuzzy rule. The number of fuzzy rules equals to the number of support vectors;

2) Given the ith support vector x_s^i ; i=1,…, L

a) The premise part of the ith fuzzy rule is evaluated as follows: the MF of fuzzy set for the jth input variable in the ith rule is

$$A_i^j(x_j) = a^j(x_j - m_i^j) \quad (10)$$

Where m_i^j is the jth element of the ith support vectorx_s^i.

b) The consequent part of the ith fuzzy rule is induced from α_0 and class labels, i.e., the consequent value of the ith rule is

$$b_i = \alpha_0^{(i)} y_s^{(i)} \quad (11)$$

Where $\alpha_0^{(i)}$ represents non-zero $\alpha_0^{(i)}$ and $y_s^{(i)}$ is the class label corresponding to the ith support vector x_s^i .Th class I membership function of x is defined using the minimum operator for

$$m_i(x) = \min_{j=1,n} m_{ij}(x) \quad (12)$$

If x is satisfied

$$n D_k(x) \begin{cases} \geq 0 \text{ for } k = t \\ \leq 0 \text{ for } k \neq t, k = 1,..n \end{cases} \quad (13)$$

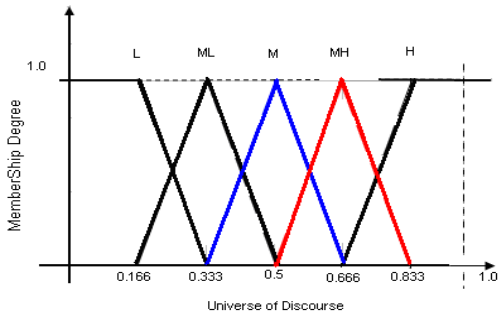

Figure 1. Membership Degree

The fuzzy rule selection procedure is described by the following steps.

Step 1) Evaluate the misclassification rates (MRs) of the rules on the validation dataset and the test dataset separately, which are represented as Evand Et.

Step 2) Set s=1 and assign a small value to threshold $h_s(h_s \geq 0)$

Step 3) Select the most influential fuzzy rules by

$$\left\{ Rule_i \middle| \alpha_0^{(i)} \text{ or } w_i \geq h_s \right\} \quad (14)$$

Step 4) Construct a fuzzy classifier (FC) by using the influential fuzzy rules selected in Step 3.

Step 5) Apply FC to the validation dataset v and the test dataset t to obtain new MRs:Ev(s).

Step 6) If Ev(s) =Ev(0) , stop the selection and use FC(s-1) as the final compact classifier and Et(s-1)as the measure of generalization performance for FC(s-1) ; Otherwise, increase s by 1, assign a higher value to threshold h_s , and go to Step 3.

The training of data is carried out by making use of fuzzy based SVM. This fuzzy based SVM is capable enough to run even when there is less number of attributes. The proposed FSVM with wiener filter is capable enough to identify and mark the attack. A salient feature of FSVM can be trained with the new dataset in regular fashion regularly.

The unique feature of the proposed classifier is that by identifying new attacks without any difficulty by generating fuzzy rule automatically. The fuzzy rules are self-generated depending upon the rule's firing strength and decision function. Then fuzzy rules are ranked based up on the impact / importance of induced fuzzy rules. These fuzzy rules are aimed to spawn a more feasible fuzzy classifier based on R values of fuzzy rules. It is to be noted that the proposed classifier need not consider all attributes that are present in the network packet .One remarkable property of SVM is its ability to learn can be independent of the feature space dimensionality which means SVM can generalize well in the presence of many features.

$$f(x) = sig\left(\sum_{i=1}^{m} y_t \alpha_i^n (x_t, x) + {}_w g^n\right) \quad (15)$$

Depending on the vector and decision taken from the rool pool, FSVM-WF classifies the attribute as attack or. If the system is not capable enough to recognize / identify the parameter is set. By the above said manner, new fuzzy rules are generated. On each time the training process will be repeated and identification is performed. The proposed mechanism contains better scope for reducing false alarm and increase the detection rate. The effect of such a sequence of layers is that the anomalous events are identified and blocked as soon as they are detected.

IV. EXPERIMENTS

In general for analyzing the performance of the classifier in IDS, the KDD Cup 1999 dataset is implemented in our research work [26], [27]. KDD Cup 1999 dataset group is obtained from the replicated raw TCP deposited data. These deposited data are collected over a timing of nine weeks over a local area Network. The tutoring data was processed to about five million connections records from seven weeks of network congestion and two weeks of testing data produced around two million connection

records. The tutoring data is composed up of 22 different attacks out of the 39 present in the test data. The well known attack types are those present in the tutoring dataset while the novel attacks are the supplementary attacks in the test datasets not available in the tutoring data sets. The attacks types are grouped into four categories:

1) **DOS:** Denial of service – e.g. syn flooding
2) **Probing:** Surveillance and other probing, e.g. port scanning
3) **U2R:** unauthorized access to local super user (root) privileges, e.g. buffer overflow attacks.
4) **R2L:** unauthorized access from a remote machine, e.g. password guessing

The training dataset consisted of 4,94,021 records among which 97,277 (19.69%) were normal, 3,91,458 (79.24%) DOS, 4,107 (0.83%) Probe, 1,126 (0.23%) R2L and 52 (0.01%) U2R connections. In each connection are 41 attributes describing different features of the connection and a label assigned to each either as an attack type or as normal. Simulation results shows that the proposed FSVM-WF attains better detection rate and reduced false alarm rate.

Table 1. Detection Rate

	DoS	Probe	U2R	R2L
FSVM-WF	96.9	97.1	96.3	58
ELM with Semantic Feature [27]	96.8	75	5.3	4.2
PSO with SVM [26]	97.9	98.6	68.9	19.5

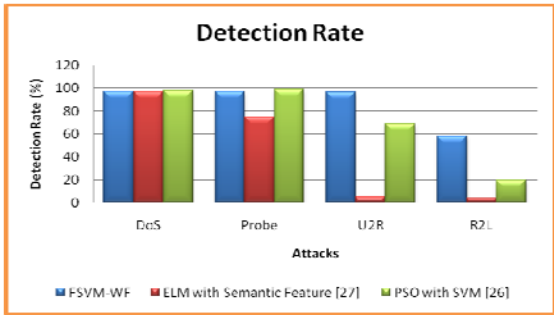

Figure 2. Detection Rate

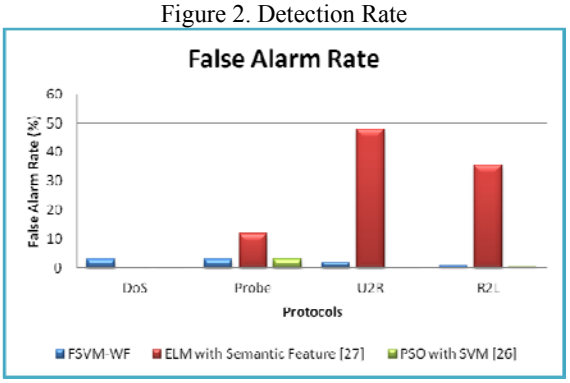

Figure 3. False Alarm Rate

Table II. False Alarm Rate

	DoS	Probe	U2R	R2L
FSVM-WF	3.04	2.95	2.032	0.6
ELM with Semantic Feature [27]	0.1	11.7	47.8	35.4
PSO with SVM [26]	0.07	3.1	0.05	0.35

V. CONCLUSION

The proposed research work intends in design and development of fuzzy based support vector machine classifier with wiener filter (FSVM – WF) for intrusion detection system. The proposed research work performs preprocessing using wiener filter. The fuzzy logic system is accommodated with support vector machine for performing the classification task. KDD cup'99 dataset that contains four major types of attacks in the network is chosen for performing FSVM-WF classification. Performance metrics detection rate and false alarm rate are chosen. Simulation results depicts that the proposed FSVM-WF classifier outperforms in terms of improved detection rate and reduced false alarm rate.

VI. REFERENCES

[1] Overview of Attack Trends, attack_trends.pdf, 2002.

[2] R.Karthik and B.L Shivakumar, "A Taxonomy of Network Intrusion Detection System for Wireless Communication", International Journal of Computer Sciences and Engineering, vol. 3, pp. 35-42, issue-12, E-ISSN: 2347-2693, December 2015.

[3] K.K. Gupta, B. Nath, R. Kotagiri, and A. Kazi, "Attacking Confidentiality: An Agent Based Approach," Proc. IEEE Int'l Conf. Intelligence and Security Informatics (ISI '06), vol.3975, pp. 285-296, 2006.

[4] J Anderson, "An Introduction to Neural Networks" (MIT, Cambridge, 1995).

[5] B Rhodes, J Mahaffey, and J Cannady, "Multiple self-organizing maps for intrusion detection", Paper presented at the Proceedings of the 23rd National Information Systems Security Conference, Baltimore, pp.16–19, 2000.

[6] A. Sung, and S. Mukkamala, "Identifying important features for intrusion detection using support vector machines and neural networks", in *Symposium on Applications and the Internet*, pp. 209–216. 2003.

[7] K.K. Gupta, B. Nath, and R. Kotagiri, "Network Security Framework," Int'l J. Computer Science and Network Security, vol. 6, no. 7B, pp. 151-157, 2006.

[8] W. Lee and S. Stolfo, "Data Mining Approaches for Intrusion Detection," Proc. Seventh USENIX Security Symp. (Security '98), pp. 79-94, 1998.

[9] W. Lee, S. Stolfo, and K. Mok, "Mining Audit Data to Build Intrusion Detection Models," Proc. Fourth Int'l Conf. Knowledge Discovery and Data Mining (KDD '98), pp. 66-72, 1998.

[10] W. Lee, S. Stolfo, and K. Mok, "A Data Mining Framework for Building Intrusion Detection Model," Proc. IEEE Symp. Security and Privacy (SP '99), pp. 120-132, 1999.

[11] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. ACM SIGMOD, vol. 22, no. 2, pp. 207-216, 1993.

[12] T. Abraham, IDDM: Intrusion Detection Using Data Mining Techniques, 2008.

[13] L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion Detection with Unlabeled Data Using Clustering," Proc. ACM Workshop Data Mining Applied to Security (DMSA), 2001.

[14] H. Shah, J. Undercoffer, and A. Joshi, "Fuzzy Clustering for Intrusion Detection," Proc. 12th IEEE Int'l Conf. Fuzzy Systems (FUZZ-IEEE '03), vol. 2, pp. 1274-1278, 2003.

[15] N.B. Amor, S. Benferhat, and Z. Elouedi, "Naive Bayes vs. Decision Trees in Intrusion Detection Systems," Proc. ACM Symp. Applied Computing (SAC '04), pp. 420-424, 2004.

[16] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur, "Bayesian Event Classification for Intrusion Detection," Proc. 19th Ann. Computer Security Applications Conf. (ACSAC '03), pp. 14-23, 2003.

[17] S. Forrest, S.A. Hofmeyr, A. Somayaji, and T.A. Longstaff, "A Sense of Self for Unix Processes," Proc. IEEE Symp Research in Security and Privacy (RSP '96), pp. 120-128, 1996.

[18] Y. Du, H. Wang, and Y. Pang, "A Hidden Markov Models-Based Anomaly Intrusion Detection Method," Proc. Fifth World Congress on Intelligent Control and Automation (WCICA '04), vol. 5, pp. 4348-4351, 2004.

[19] W. Wang, X.H. Guan, and X.L. Zhang, "Modeling Program Behaviors by Hidden Markov Models for Intrusion Detection," Proc. Int'l Conf. Machine Learning and Cybernetics (ICMLC '04), vol. 5, pp. 2830-2835, 2004.

[20] C. Warrender, S. Forrest, and B. Pearlmutter, "Detecting Intrusions Using System Calls: Alternative Data Models," Proc. IEEE Symp. Security and Privacy (SP '99), pp. 133-145, 1999.

[21] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," Proc. 18th Int'l Conf. Machine Learning (ICML '01), pp. 282-289, 2001.

[22] N Jaisankar, SGP Yogesh, A Kannan, K Anand, Intelligent Agent Based Intrusion Detection System Using Fuzzy Rough Set Based Outlier Detection, Soft Computing Techniques in Vision Sci., SCI 395 (Springer, 2012), pp. 147–153.

[23] Vahid Golmah, An Efficient Hybrid Intrusion Detection System based on C5.0 and SVM, International Journal of Database Theory and Application vol.7, No.2, pp.59-70 (2014),.

[24] Zhai Jinbiao, "Research on Intrusion Detection System Based on Clustering Fuzzy Support Vector Machine", International Journal of Security and Its Applications vol.8, No.3 , pp. 249-260, (2014).

[25] Md. Al Mehedi Hasan, Mohammed Nasser, Biprodip Pal and Shamim Ahmad, "Support Vector Machine and Random Forest Modeling for Intrusion Detection System (IDS)", Journal of Intelligent Learning Systems and Applications, No.6, pp.45-52, 2014.

[26] Jashan Koshal and Monark Bag, "Cascading of C4.5 Decision Tree and Support Vector Machine for Rule Based Intrusion Detection System", I. J. Computer Network and Information Security, No.8, pp. 8-20, 2012.

[27] B. Sujitha, and V. Kavitha, "Layered Approach For Intrusion Detection Using Multiobjective Particle Swarm Optimization", International Journal of Applied Engineering Research, vol.10, No.12, pp. 31999 – 32014, 2015.

[28] G. Creech and J. Hu, "A Semantic Approach to Host-Based Intrusion Detection Systems Using Contiguous and Discontiguous System Call Patterns", IEEE Transactions on Computers, vol. 63, No. 4, pp. 807 – 819, 2014.

