Anomaly Traffic Detection Based on PCA and SFAM

Preecha Somwang[1, 2] and Woraphon Lilakiatsakun[2]
[1]Office of Academic Resources and Information Technology, Rajamangala University of Technology Isan, Thailand
[2]Faculty of Information Science and Technology, Mahanakorn University of Technology, Thailand

Abstract: *Intrusion Detection System (IDS) has been an important tool for network security. However, existing IDSs that have been proposed do not perform well for anomaly traffics especially Remote to Local (R2L) attack which is one of the most concerns. We thus propose a new efficient technique to improve IDS performance focusing mainly on R2L attacks. The Principal Component Analysis (PCA) and Simplified Fuzzy Adaptive resonance theory Map (SFAM) are used to work collaboratively to perform feature selection. The results of our experiment based on KDD Cup'99 dataset show that this hybrid method improves classification performance of R2L attack significantly comparing to other techniques while classification of the other types of attacks are still well performing.*

Keywords: *IDS, network security, PCA, SFAM.*

Received May 3, 2013; accepted July 22, 2013; published online June 26, 2014

1. Introduction

For years, intrusion attacks [11] have made great damages of computer system. So, intrusion detection techniques have been interesting topics in the network security. The main idea is how to distinguish and predict normal and abnormal behaviours. Generally, there are two main approaches of intrusion detection technique which are namely misuse detection and anomaly detection, as shown in Figure 1.

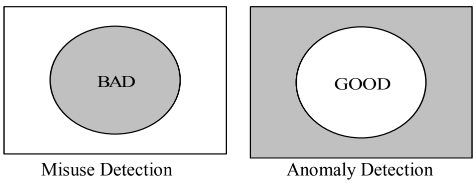

Figure 1. Misuse detection and anomaly detection concept.

Misuse detection is based on predefined signature of known intrusion to match with monitored traffics. Contrastingly, anomaly detection first creates the normal profile that contains metrics derived from the system operation and then current observation will be compared with the normal profile in order to detect change in the patterns of utilization or behaviour of the system [31].

However, major problem in intrusion detection is that new attacks cannot be detected by misuse detection technique due to no predefined signatures to match the observed traffics. As a result, anomaly detection plays an important role to detect the intrusion in computer network system. To perform anomaly detection, various techniques have been widely applied as supervised, semi-supervised and unsupervised technique [7].

Nevertheless, existing techniques do not well perform for Remote to Local (R2L) or outsider's attack [23]. It is because that R2L attack offers the most assorted set of attacks in terms of attack execution, implementation and dynamics. We thus propose the new anomaly detection technique mainly focusing on R2L attack that applied supervised anomaly detection learning technique. It combines the PCA used for random selection of the best attribution and SFAM used for classifying different group of normal and abnormal data.

The rest of this paper is organized as following. Section 2 discusses background and related works of anomaly detection. Section 3 explains the KDD Cup'99 dataset. Section 4 describes the methodology, PCA and SFAM. Experiment and results are shown in sections 5 and 6 consequently. Finally, we conclude this article in section 7.

2. Background and Related Works

2.1. Background

Many techniques of anomaly detection have been proposed and categorized as supervised learning technique, semi-supervised learning technique and unsupervised learning technique.

- *Supervised Learning Technique* [10]: Is the use of training data consisting of instances which are labelled as both normal and anomaly class. These instances are used to train on models. The typical approach for this technique is to build a predictive model for the normal and anomaly class. Detection is based on the characteristics of known attacks, called signature, any actions that match with any signatures are considered as intrusive. The advantage of supervised learning technique is that it can perform well to detect known malicious attacks. However, it could generate high false alarm rate of new attacks without signatures [27].

- *Semi-supervised Learning Technique* [29]: Is the use of the training data consisting of instances which are labelled only normal class, no anomaly class required. The approach used in such technique is to build a predictive model for the class corresponding to normal behavior. Therefore, any action that significantly deviates from the normal behaviour is considered as intrusive action. The advantage of semi-supervised technique is that it can detect unknown and known type of attack. But, the limitation is that it is difficult to obtain dataset which represents all possible normal behaviour [25].
- *Unsupervised Learning Technique*: Does not require labelled training data. In [22] they use data processing on Distance Based Outlier Detection (DBOD). While develop classification technique by comparing between test pattern and stored normal patterns. Mazal *et al*. [14] proposed a new technique called Inter-Clustering Result Association (ICRA) to improve robustness and correctness of the decision making process. However, unsupervised learning technique still cause significant false alarm since models describing complete normal behaviours are very difficult to obtain.

2.2. Related Works

For learning process [16], supervised learning technique is efficient to build classifiers. As previously mentioned, it can take advantage of the known target outputs to train the classifier to perform classification. Supervised learning method based on support vector machine was proposed by Yang *et al*. [30]. The results showed the high detection rate whereas low false alarm rate, but there are some crucial problems on selects of the best attribution and reduction the feature space. Then, such problems can be resolved by using PCA technique. Nziga and Cannady [18] proposed a hybrid feature selection method based on Mutual Information Difference evaluation criteria and Principal Component Analysis (MID-PCA) algorithm to improve efficiency of selects the best attribution on KDD Cup'99 dataset. Terrence [24] applied genetic algorithm to feature subset of data for generating fuzzy rule. Then, fuzzy logic is applied to calculate the fitness function used to define the normal or abnormal behavior of network system. Results show that performance of such technique could reduce the false alarm rate. But, PCA does not scale well with complexity. As a result, the stop criterion does not clear in every situation.

Li [12] proposed the neural network classifier including two parts of process. The first part used 41 features for training data and second part classified data by using 3 layers feed-forward neural network model. Mukhopadhyay *et al*. [17] neural network used KDD Cup'99 dataset to test the feasibility of this model. These techniques showed better effectiveness of detection for attacks and also yielding false alarm rate.

Finally, the crucial problem of intrusion detection techniques has still been left. All proposed techniques cannot perform efficiently on R2L attacks that do not have known signatures of intrusion. Because R2L attack is dynamic properties of intrusion behaviors of unauthorized access from a remote machine of outsider's attack [28]. We thus propose the novel anomaly detection that is based on supervised learning technique by using combination of normal and anomalous behaviour to train data of various anomaly attacks.

3. KDD Cup'99 Dataset

We use a dataset from KDD Cup'99 intrusion detection as the raw data. This dataset is used for building the classification models by supervised training and for performance evaluation by validating and testing the results of the framework.

All features of a connection in the dataset are listed in the Table 1. Each connection record contains 7 discrete and 34 continuous features for a total of 41 features. We used this dataset in the experiments because it is the most comprehensive dataset that is still widely used to compare and benchmark the performance of intrusion detection models [2].

Table 1. The feature in KDD Cup'99 dataset [2].

No	Variable Name	Type	No	Variable Name	Type
1	Duration	Continuous	22	Is_guest_login	discrete
2	Protocol_type	Discrete	23	Count	Continuous
3	Service	Discrete	24	Srv_count	Continuous
4	Flag	Discrete	25	Serror_rate	Continuous
5	Src_bytes	Continuous	26	Srv_serror_rate	Continuous
6	Dst_bytes	Continuous	27	Rerror_rate	Continuous
7	Land	Discrete	28	Srv_rerror_rate	Continuous
8	Wrong_fragment	Continuous	29	Same_srv_rate	Continuous
9	Urgent	Continuous	30	Diff_srv_rate	Continuous
10	Hot	Continuous	31	Srv_diff_host_rate	Continuous
11	Num_failed_logins	Continuous	32	Dst_host_count	Continuous
12	Logged_in	Discrete	33	Dst_host_srv_count	Continuous
13	Num_compromised	Continuous	34	Dst_host_same_srv_rate	Continuous
14	Root_shell	Continuous	35	Dst_host_diff_srv_rate	Continuous
15	Su_attempted	Continuous	36	Dst_host_same_src_port_rate	Continuous
16	Num_root	Continuous	37	Dst_host_srv_diff_host_rate	Continuous
17	Num_file_creations	Continuous	38	Dst_host_serror_rate	Continuous
18	Num_shells	Continuous	39	Dst_host_srv_serror_rate	Continuous
19	Num_access_files	Continuous	40	Dst_host_rerror_rate	Continuous
20	Num_outbound_cmds	Continuous	41	Dst_host_srv_rerror_rate	Continuous
21	Is_host_login	Discrete	42	Normal or Attack	Discrete

The dataset has 41 attributes for each connection record plus one class label. There are 24 attack types, but we treat all of them as an attack group. A dataset of size N is processed. The nominal attributes are converted into linear discrete values (integers). After eliminating labels, the dataset is described as a matrix X, which has N rows and m=41 columns (attributes). There are m_d=8 discrete-value attributes and m_c=33 continuous-value attributes.

A complete list of the set of features defined for the connection records is given in the four tables, basic features, content features, traffic features and host-based features table. Table 2 shows information for the basic features of 9 individual features of TCP connections.

Table 2. Basic features of individual TCP connections.

No	Feature Name	Description
1	Duration	Length (number of seconds) of the connection
2	Protocol_type	Type of the protocol, e.g. tcp, udp, etc.
3	Service	Network service on the destination, e.g., http, telnet, etc.
4	Flag	Normal or error status of the connection
5	Src_bytes	Number of data bytes from source to destination
6	Dst_bytes	Number of data bytes from destination to source
7	Land	1 if connection is from/to the same host/port; 0 otherwise
8	Wrong_fragment	Number of ``wrong'' fragments
9	Urgent	Number of urgent packets

Table 3 shows information for the content features within a connection suggested by domain knowledge.

Table 3. Content features by domain knowledge.

No	Feature Name	Description
10	Hot	Number of ``hot'' indicators
11	Num_failed_logins	Number of failed login attempts
12	Logged_in	1 if successfully logged in; 0 otherwise
13	Num_compromised	Number of ``compromised'' conditions
14	Root_shell	1 if root shell is obtained; 0 otherwise
15	Su_attempted	1 if ``su root'' command attempted; 0 otherwise
16	Num_root	Number of ``root'' accesses
17	Num_file_creations	Number of file creation operations
18	Num_shells	Number of shell prompts
19	Num_access_files	Number of operations on access control files
20	Num_outbound_cmds	Number of outbound commands in an ftp session
21	Is_hot_login	1 if the login belongs to the ``hot'' list; 0 otherwise
22	Is_guest_login	1 if the login is a ``guest''login; 0 otherwise

The data schema of the traffic features computed using a two-second time window, as shown in Table 4.

Table 4. Traffic features.

No	Feature Name	Description
23	Count	Number of connections to the same host as the current connection in the past two seconds
24	Srv_count	Number of connections to the same service as the current connection in the past two seconds
25	Serror_rate	% of connections that have ``SYN'' errors, S0 error rate
26	Srv_serror_rate	% of connections that have ``SYN'' errors, S0 error rate for the same service as the current one
27	Rerror_rate	% of connections that have ``REJ'' errors, RST error rate
28	Srv_rerror_rate	% of connections that have ``REJ'' errors, RST error rate for the same service as the current one
29	Same_srv_rate	% of connections to the same service
30	Diff_srv_rate	% of connections to different services
31	Srv_diff_host_rate	% of connections to different hosts

Table 5 shows information for the host-based features from the communication of source address to destination address connection.

Table 5. Host-based features.

No	Feature Name	Description
32	Dst_host_count	Count of connections having the same destination.
33	Dst_host_srv_count	Count of connections having the same destination host and using the same service.
34	Dst_host_same_srv_rate	% of connections having the same destination host
35	Dst_host_diff_srv_rate	% of different services on the current host.
36	Dst_host_same_src_port_rate	% of connections to the current host having the same src port.
37	Dst_host_srv_diff_host_rate	% of connections to the same service coming from different host.
38	Dst_host_serror_rate	% of connections to the current host that have an S0 error.
39	Dst_host_srv_serror_rate	% of connections to the current host and specified service that have an S0 error
40	Dst_host_rerror_rate	% of connections to the current host that have an RST error.
41	Dst_host_srv¬_rerror_rate	% of connections to the current host and specified service that an RST error.

There are 4 attacked class types of IDS of this experimental model, presented in the Table 6 [4].

Table 6. Data attack type [4].

Class	Known Attack	Unknown Attack
DoS	back, land, neptune, pod, smurf, teardrop	apache2,mailbomb, processtable, udpstorm
Probe	ipsweep, nmap, portsweep, satan	mscan, saint
U2R	buffer_overflow,loadmodule, perl,rootkit	ps, sqlattack, xterm
R2L	ftp_write, guess_passwd, phf, imap,multihop, warezmaster, Warezclient	httptunnel, named, sendmail, snmpgetattack, snmpguess, worm,xlock,xsnoop

- *Denial of Service* (*DoS*): Such as ping of death, attackers take a computing or memory resource too busy to handle legitimate requests. Thus, denying legitimate users access to a machine.
- *Probing* (*Probe*): Such as port scanning attack, attacker scans a computer network to gather information or find know vulnerabilities.
- *User to Root* (*U2R)*: Unauthorized access to local root privileges, attacker starts out with access to normal user account on the system and is able to exploit vulnerability to gain root access to the system.
- *R2L*: Unauthorized access from the remote machine, where an attacker sends packets to a machine over a network. Then exploits the machine's vulnerability to illegally gain local access as a user.

The reason behind using anomaly detection is that like R2L attack, its outsider's attack also diverse in nature and have high false positive rate.

4. Proposed Method

We developed a new framework based on 3 major steps, as shown conceptually in Figure 2. The first step is data pre-processing that handles missing and incomplete data. The second step is to do feature selection by PCA algorithm. The last step is to classify different group of normal and anomalous data by SFAM algorithm.

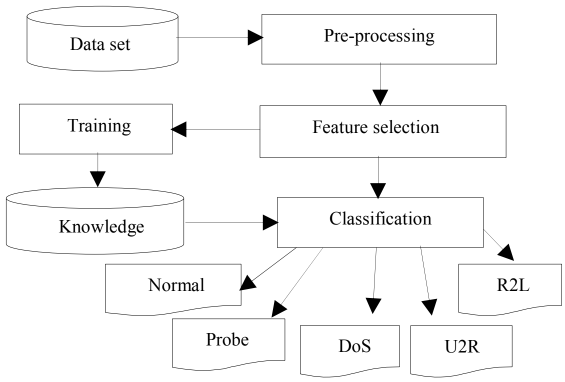

Figure 2. Overall architecture of system.

4.1. Data Pre-Processing

Data pre-processing is the process of cleansing incomplete data of involved mapping symbolic-valued attributes to numeric-valued attributes. This process is implemented non-zero numerical features of variables for intrusion detection dataset [6].

Each record captures various connection features, such as protocol_type, threr are 3 different symbols, tcp, udp and icmp, presented in the Table 7.

Table 7. Mapping feature No. 2 (protocol type).

Field Name	Value
Tcp	1
Udp	2
Icmp	3

The feature No. 3 is a service of network service on the destination (68 symbols), presented in the Table 8.

Table 8. Mapping feature No. 3 (Service).

Field Name	Value	Field Name	Value	Field Name	Value	Field Name	Value
Ftp	1	telnet	18	Bgp	35	Gopher	52
Private	2	Ntp_u	19	Ldap	36	Hostname	53
Name	3	Remote_job	20	Uucp	37	Iso_tsap	54
Domain	4	link	21	Netstat	38	Klogin	55
Daomain_U	5	Pop_3	22	Kshell	39	Netbios_dgm	56
Http	6	Tftp_u	23	Sql_net	40	Netbios_ns	57
Smtp	7	Urp_i	24	Netbio_ssn	41	Pm_dump	58
Ftp_Data	8	Tim_i	25	http_443	42	Rje	59
Icmp	9	Login	26	Whois	43	Ssh	60
Other	10	Imap4	27	Courier	44	Sunrpc	61
Eco_I	11	Pop_2	28	Nnsp	45	Supdup	62
Auth	12	Vmnet	29	Csnet_ns	46	Systat	63
Ecr_I	13	Shell	30	Ctf	47	Uucp_path	64
Irc	14	prinetr	31	Daytime	48	Z39_50	65
X11	15	nntp	32	Discard	49	Netbios_ssn	66
Finger	16	echo	33	Efs	50	Urh_i	67
Time	17	mtp	34	Exec	51	Red_i	68

The feature No. 4 is status of the connection (flag), normal or error connection. There are 11 different symbols, presented in the Table 9.

Table 9. Mapping feature No. 4 (Flag).

Field Name	Value	Field Name	Value
SF	1	S3	7
RSTR	2	RSTOSO	8
S0	3	RSTO	9
S1	4	SH	10
S2	5	OTH	11
REJ	6		

There are 5 describes of the taxonomy of normal or attacks behavior in feature No. 42 (type). They are normal (group #1), DoS (group #2), Probe (group #3), U2R (group #4) and R2L (group #5) which as shown in Table 10.

Table 10. Mapping feature No. 42 (Type).

Field Name	Value	Group	Field Name	Value	Group
Normal	1	1	Ps	21	4
Apache2	2	2	Rootkit	22	4
Back	3	2	Sqlattack	23	4
Land	4	2	Xterm	24	4
Mailbomp	5	2	Ftp_Write	25	5
Neptune	6	2	Guess_Passwd	26	5
Pod	7	2	Httptunnel	27	5
Processtable	8	2	Imap	28	5
Smurf	9	2	Multihop	29	5
Teardrop	10	2	Named	30	5
Udpstorm	11	2	Phf	31	5
Ipsweep	12	3	Sendmail	32	5
Mscan	13	3	Snmpgetattack	33	5
Nmap	14	3	Snmguess	34	5
Portsweep	15	3	Warezmaster	35	5
Saint	16	3	Worm	36	5
Satan	17	3	Xlock	37	5
Buffer_Overflow	18	4	Xsnoop	38	5
Loadmodue	19	4	Warezclient	39	5
Perl	20	4	Spy	40	5

Each feature symbol is mapped to integer values ranging from 1 to N where, N is the number of symbols. Features having value ranges like duration [0, 58329], num_compromised [0,884], count [0, 511], dst_host_count [0, 255], src_bytes [0, 693375640], dst_bytes [0, 5203179] were scaled linearly to the range [0.0, 1.0] defined as Equation 1:

$$x = \frac{x - min}{max - min} \tag{1}$$

Given x= feature value, min=minimum value, max= maximum value of value ranges.

4.2. Feature Selection

Feature selection is the process of selecting a subset of relevant features for use in model construction. Given the benchmark data from KDD Cup'99 dataset, which is an original complete feature composed of 41 attributes for PCA selecting the best attribution and reducing of feature space. We reduce the dimensionality of this dataset 21 features were selected out of 41 features as following Field No. 1, 2, 5, 6, 7, 8, 9, 10, 11, 14, 16 , 17, 18, 19, 23, 25, 28, 30, 31, 32, and 33, for training process, namely duration, protocol_type, flag, dst_bytes, land, wrong_fragment, urgent, num_failed_logins, root_shell, num_root, num_file_creations, num_shells, num_access_files, count, serror_rate, srv_rerror_rate, diff_srv_rate, srv_diff_host_rate, dst_host_count and dst_host_ srv_count [15].

PCA has been proposed as a method of traffic anomaly detection, its application very popular in the networking community. PCA is a powerful tool for analyzing data of patterns in data can be hard to find in data of high dimension. Aim to reducing the number of dimensions, without much loss of information [1]. PCA is an optimal linear dimension reduction method in the sense of least mean square error. By projecting the original feature vector to a smaller subspace, PCA achieves the effect of dimension reduced and redundancy removed. Principal components are particular linear combinations of the m random variables x_1, x_2, ..., x_m calculated from the correlation matrix, the size of which scales quadratic ally with the number of variables, m [5].

Given the KDD Cup'99 dataset has been 41 features represented by x_1, x_2, ..., x_{41} where each observation is represented by a vector of length m, the dataset is represented by a matrix x_{nxm} in the Equation 2 [3].

$$x_{nxm} = \begin{bmatrix} x_{11}, \ldots, x_{1m} \\ x_{21}, \ldots, x_{2m} \\ \ldots, \ldots, \ldots \\ x_{n1}, \ldots, x_{nm} \end{bmatrix} = \left[x_1, \ldots, x_n \right] \tag{2}$$

The average observation of training set using Equation 3:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{3}$$

The deviation from the average is defined as Equation 4:

$$\Phi_i = x_i - \mu \tag{4}$$

The sample covariance matrix of the dataset is defined as Equation 5:

$$C = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \mu\right)\left(x_i - \mu\right)^T = \frac{1}{n}AA^T \tag{5}$$

In PCA the covariance matrix has large dimension, therefore the computation of eigenvectors is time consuming and the results are not always satisfactory. The eigenvalues and eigenvectors are then calculated from the covariance matrix: x=x_i, x_2, ..., x_m to be normalized. Suppose (λ_1, μ_1), (λ_2, μ_2), (λ_m, μ_m) are m eigenvalue-eigenvector pairs of the sample covariance matrix C. The dimensionality of the subspace k can be determined, as shown in the Equation 6 [9].

$$\frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{m}\lambda_i} \geq \alpha \tag{6}$$

The dataset contains training data that previously began with more than 300,000 records, as shown in Table 11.

Table 11. Training dataset.

Type	Sub Type	Amount	%
Normal		66,395	21.396
DoS	Pod	206	0.066
	Smurt	154,901	49.918
	Back	1,098	0.354
	Land	19	0.006
	Neptune	58,001	18.691
	Teardrop	918	0.296
Probe	Ipsweep	3,723	1.200
	Portsweep	3,564	1.149
	Nmap	1,554	0.501
	Satan	5,019	1.617
U2R	Buffer_Overflow	30	0.010
	Loadmodule	13	0.004
	Perl	6	0.002
	Rootkit	21	0.007
R2L	Guess_Passwd.	9,720	3.132
	Multihop	18	0.006
	Phf.	4	0.001
	Ftp_Write.	8	0.003
	Imap.	12	0.004
	Spv.	2	0.001
	Warezclient.	2,613	0.842
	Warezmaster.	2,468	0.795
Total		**310,313**	**100**

We random selected to approximately 18, 216 records for testing presented in the Table 12.

Table 12. Dataset for attack distribution testing

Attack Type	Population Size
Normal	5,763
DoS	3,530
Probe	2,164
U2R	70
R2L	6,689
Summary	18,216

4.3. Classification

We study the performance of our proposed scheme of classifier by SFAM. It's a simplified version of the fuzzy ARTMAP neural network model. It was designed to improve the computational efficiency of the fuzzy ARTMAP model with a minimal loss of learning effectiveness, as shown construction in Figure 3 [26].

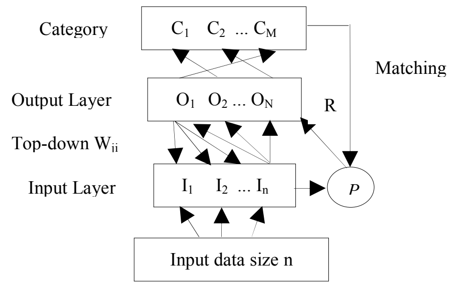

Figure 3. The SFAM network architecture [26].

The input vectors are first complement code to become vectors I which are applied to the input layer. Each node in the output category layer is linked through a set of top-down weights to each node in the input layer. The N nodes in output category layer label the M category or class that the SFAM has to learn to recognize. Usually, N>M when active during testing an output category node indicates the class by pointing to the corresponding category classification node. The vigilance parameter ρ has to be chosen to determine the number of classes found. Match tracking causes automatic adjustment of ρ if classification errors are found in training [20].

The choice parameter is $\alpha>0$, learning rate parameter is $\beta \in [0,1]$, vigilance parameter is $\rho \in [0,1]$, weight vector is w_{ji}. Once SFAM has been trained, a feed forward pass through the compliment-code and the input layer classifies an unknown pattern.

The SFAM operation is defined as:

- *Step1*: Initialize network weights and parameters w_{ji}, α, β, ρ. Set w_{ji}=1, j=1, 2, ..., M, I= 1, 2, ..., N Select values for parameters: $\alpha > 0, \beta \in [0,1]$, and $\rho \in [0,1]$.
- *Step 2*: Read Scaled input I.
- *Step 3*: For Every output node j, compute, shown in the Equation 7:

$$T_j(I) = \frac{\left|I \wedge w_j\right|}{\alpha + \left|w_j\right|} \tag{7}$$

For nodes j=1, 2, ..., M, where '$\wedge$' is the fuzzy AND operator defined as $(x \wedge y)_i$= $min(x_i, y_i)$ and the norm $|.|$ is defined by the Equation 8:

$$|x| = \sum_{i=1}^{M}\left|x_i\right| \tag{8}$$

- *Step 4*: Select output node whose exemplar matches with input best, Best matching exemplar, shown in the Equation 9:

$$T_j = \max\left\{T_j : j = 1, 2, ..., M\right\} \tag{9}$$

The degree of match between the output category node and an input vector is given by the match function, $MF(I, w_j)$ defined by the Equatuhion10:

$$MF(I, w_j) = \frac{I \wedge w_j}{|I|} = \frac{|I \wedge w_j|}{d} \quad (10)$$

- *Step 5*: Check if this match is within specified similarity level: Resonance test (degree of similarity with best matching exemplar), shown in the Equation 11:

$$\frac{|I \wedge w_j|}{I} \geq \rho \quad (11)$$

If similar go to Step 7.
Else go to next Step 6.

- *Step 6*: Enable selection of a new output node and exemplar for this input: mismatch reset:
Set T_J=1 and go to Step 4.
- *Step 7*: Update best-matching exemplar (learning law) shown in the Equation 12:

$$w_j^{(new)} = \beta \left(I \wedge w_j^{(old)} \right) + (1 - \beta) w_j^{(old)} \quad (12)$$

- *Step 8*: go to Step 2 to read the next input [21].

5. Experimental

The proposed method and the other techniques were simulated on Microsoft windows XP operating system by using MATLAB toolbox. The parameters considered in the evaluation phase are: the number of clusters in SFAM neural net, the number of epochs in the training phase of SFAM neural net and the vigilance parameter, presented in Table 13.

Table 13. Parameters of the proposed method.

No. of epochs (time)	Vigilance Parameter (ρ)	Choice Parameter (α)	Learning Rate (β)
100	0.65	0.000001	1

The effect of these parameters has been evaluated based on the normal generalization, intrusive generalization, overall generalization, discrimination ability, FP and FN as describes ahead, as shown in the Equations 13 and 14, and presented in the Table 14 [8].

$$DetectionRate = \frac{TP}{TP + FN} \quad (13)$$

$$FalseAlarmRate = \frac{FP}{FP + TN} \quad (14)$$

Table 14. Confusion for evaluation of attack.

Type	Predicted Connection	
	Attack	Normal
Attack	True Positive (TP)	False Negative (FN)
Normal	False Positive (FP)	True Negative (TN)

Detection rate is computed as the ratio between the numbers of correctly detected TP attacks and the total number of attacks. FP Rate is computed as the ratio between the numbers of normal connections that are incorrectly misclassified as attacks. The performance of classifiers is evaluated with respect to their classification of unseen normal and intrusive patterns.

6. Results

We use accuracy of detection (Detection rate) and error of detection (false alarm rate) as performance metrics. We compare our method with Artificial Neural Network (ANN) by Poojitha *et al*. [19] and three-level hybrid methods of Decision Tree, Naïve Bayes and Baysian Clustering by Lu and Xu [13]. Table 15 presents the results of ANN and three-level hybrid methods and Table 16 show the performance of the proposed method.

Table 15. Performance of ANN and Three-level hybrid method.

Class Type	ANN [19]		Three-level [13]	
	Detection Rate	False Alarm Rate	Detection Rate	False Alarm Rate
Normal	99.76 %	0.24 %	94.68 %	5.32 %
Probe	100 %	0 %	93.50 %	6.50 %
DoS	100 %	0 %	98.54 %	1.46 %
U2R	67.77 %	32.23 %	97.14 %	2.86 %
R2L	36.84 %	63.16 %	48.91 %	51.09 %
Average	80.87 %	19.13 %	86.55 %	13.45 %

Table 16. Performance of the proposed method.

Class type	# of Record	Hit	Miss	Detection Rate	False Alarm Rate
Normal	5,763	5,719	44	99.24 %	0.76 %
Probe	2,164	2,091	73	96.63 %	3.37 %
DoS	3,530	3,473	57	98.38 %	1.62 %
U2R	70	62	8	88.57 %	11.43 %
R2L	6,689	6,539	150	97.75 %	2.25 %
Summary	18,216	17,884	332	96.11 %	3.89 %

7. Conclusions

This article presents a new hybrid method by using PCA and SFAM to improve anomaly detection performances. Simulation results show that the proposed method outperforms the other two methods, ANN and three-level hybrid, distinctively. It provides averagely high performance of detection rate which is 96.11 % and also minimizes the false alarm rate down to 3.89 %. Moreover, this method can improve the effectiveness of detection of R2L attacks significantly comparing with the other 2 methods. Even though we can achieve our goal to improve overall performance of anomaly detection but our proposed method does not perform well for U2R. Our future work is thus to improve U2R detection along with keeping high performance of the other type of attacks and then implement our algorithm in the real life environment.

References

[1] Ahmed K., El-Henawy M., Rashad Z., and Nomir O., "On-Line Signature Verification Based on PCA Feature Reduction and Statistical Analysis," *in Proceedings of the International Conference on Computer Engineering and System*, Cairo, Egypt, pp. 3-8, 2010.

[2] Alsharafat W., "Applying Artificial Neural Network and eXtended Classifier System for Network Intrusion Detection," *the International Arab Journal of Information Technology*, vol. 10, no. 3, pp. 230-238, 2013.

[3] Bin Haji S., Abdullah H., bin Abu Bak K., bin Ngadi A., Dahlan D., and Chimphlee W., "A Novel Method for Unsupervised Anomaly Detection using Unlabelled Data," *in Proceedings of International Conference Computational Sciences and its Applications*, Perugia, pp. 252-260, 2010.

[4] Devaraju S. and Ramakrishnan S., "Performance Analysis of Intrusion Detection System using Various Neural Network Classifiers," *in Proceedings of International Conference on Recent Trends in Information Technology*, Chennai, Tamil Nadu, pp. 1033-1038, 2011.

[5] Good P., Kost D., and Cherry A., "Introducing a Unified PCA Algorithm for Model Size Reduction," *in Proceedings of IEEE Transactions on Semiconductor Manufacturing*, Austin, USA, pp. 201-209, 2010.

[6] Gou S., Wang Y., Jiao L., Feng J., and Yao Y., "Distributed Transfer Network Learning based Intrusion Detection," *in Proceedings of International Symposium on Parallel and Distributed Processing with Applications*, Chengdu, pp. 511-515, 2009.

[7] Hodge V. and Austin J., "A Survey of Outlier Detection Methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85-126, 2004.

[8] Jahanbani A. and Karim H., "A New Approach for Detecting Intrusions Based on the PCA Neural Network," *Journal of Basic and Applied Scientific Research*, vol. 1, no. 2, pp. 672-679, 2012.

[9] Jolliffe T., *Principal Component Analysis*, Springer-Verlag, New York, 2002.

[10] Khakpour N. and Jalili S., "Using Supervised and Transductive Learning Techniques to Extract Network Attack Scenarios," *in Proceedings of the 14th International CSI Computer Conference*, Tehran, pp. 71-76, 2009.

[11] Li L. and Zhao K., "A New Intrusion Detection System Based on Rough Set Theory and Fuzzy Support Vector Machine," *in Proceedings of the 3rd International Workshop on Intelligent Systems and Applications*, pp. 1-5, 2011.

[12] Li X., "Optimization of the Neural-Network-Based Multiple Classifiers Intrusion Detection System," *in Proceedings of International Conference on Internet Technology and Applications*, Wuhan, pp. 1-4, 2010.

[13] Lu H. and Xu J., "Three-level Hybrid Intrusion Detection System," *in Proceedings of International Conference on Information Engineering and Computer Science*, Wuhan, pp. 1-4, 2009.

[14] Mazal J., Casas P., Labit Y., and Owezarski P., "Sub-Space Clustering, Inter-Clustering Results Association and Anomaly Correlation for Unsupervised Network Anomaly Detection," *in Proceedings of the 7th International Conference on Network and Service Management*, Paris, pp. 1-8, 2011.

[15] Mechtri L., Djemili F., and Ghoualmi N., "Intrusion Detection using Principal Component Analysis," *in Proceedings of the 2nd International Conference on Engineering Systems Management and Its Applications*, Sharjah, pp. 1-6, 2010.

[16] Meyn S., Surana A., Lin Y., and Narayanan S., "Anomaly Detection Using Projective Markov Models in a Distributed Sensor Network," *in Proceedings of the 48th IEEE Conference on Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference*, Shanghai, pp. 4662-4669, 2009.

[17] Mukhopadhyay I., Chakraborty M., Chakrabarti S., and Chatterjee T., "Back Propagation Neural Network Approach to Intrusion Detection System," *in Proceedings of International Conference on Recent Trends in Information Systems*, Kolkata, pp. 303-308, 2011.

[18] Nziga J. and Cannady J., "Minimal Dataset for Network Intrusion Detection Systems via MID-PCA: A Hybrid Approach," *in Proceedings of the 6th International Conference Intelligent Systems*, Sofia, pp. 453-460, 2012.

[19] Poojitha G., Kumar N., and Reddy J., "Intrusion Detection using Artificial Neural Network," *in Proceedings of the 2nd International Conference on Computing, Communication and Networking Technologies*, Karur, pp. 1-7, 2010.

[20] Primekumar P. and Idiculla M., "On-Line Malayalam Handwritten Character Recognition using Wavelet Transform and SFAM," *in Proceedings of the 3ed International Conference on Electronics Computer Technology*, Kanyakumari, pp. 49-53 2011.

[21] Rajasekaran S. and Vijayalakshmi A., "Image Recognition using Simplified Fuzzy ARTMAP Augmented with a Moment based Feature Extractor," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 14, no. 8, pp. 1081-1095, 2000.

[22] Said D., Stirling L., Federolf P., and Barker K., "Data Preprocessing for Distance-Based Unsupervised Intrusion Detection," *in Proceedings of the 9th Annual International Conference on Privacy, Security and Trust*, Montreal, pp. 181-188, 2011.

[23] Tang P., Jiang R., and Zhao M., "Feature Selection and Design of Intrusion Detection System Based on k-Means and Triangle Area Support Vector Machine," *in Proceedings of the 2nd International Future Networks*, Sanya, Hainan, pp. 144- 148, 2010.

[24] Terrence F., "Evolutionary Optimization of a Fuzzy Rule-based Network Intrusion Detection System," *in Proceedings of Annual Meeting of the North American Fuzzy Information Processing Society*, Toronto, pp. 1-6, 2010.

[25] Vatanen T., Kuusela M., Malmi E., Raiko T., Aaltonen T., and Nagai Y., "Semi-Supervised Detection of Collective Anomalies with an Application in High Energy Physics," *in*

Proceedings of IEEE International Joint Conference on Neural Networks, Brisbane, pp.1-8, 2012.

[26] Venkatesan P. and Suresh M., "Classification of Renal Failure using Simplified Fuzzy Adaptive Resonance Theory Map," *International Journal of Computer Science and Network Security*, vol. 9, no. 11, pp. 129-134, 2009.

[27] Wattanapongsakorn N., Srakaew S., Wonghirunsombat E., Sribavonmongkol C., Junhom T., Jongsubsook P., and Charnsripinyo C., "A Practical Network-Based Intrusion Detection and Prevention System," *in Proceedings 11th International Conference on Trust, Security and Privacy in Computing and Communications*, Liverpool, pp. 209-214, 2012.

[28] Wu J., Chaing D., Lin T., Chung Y., and Chen T., "A Reliable Dynamic User-Remote Password Authentication Scheme over Insecure Network," *in Proceedings of the 26th International Conference on Advanced Information Networking and Applications Workshops*, Fukuoka, pp. 25-28, 2012.

[29] Xiang G. and Min W., "Applying Semi-supervised Cluster Algorithm for Anomaly Detection," *in Proceedings of the 3rd International Symposium on Information Processing*, Qingdao, pp. 43-45, 2010.

[30] Yang C., Yang H., and Deng F., "Quantum-Inspired Immune Evolutionary Algorithm based Parameter Optimization for Mixtures of Kernels and its Application to Supervised Anomaly IDSs," *in Proceedings of the 7th World Congress on Intelligent Control and Automation*, Chongqing, pp. 4568-4573, 2008.

[31] Zhong J., Wu H., and Lai Y., "Intrusion Detection using Evolving Fuzzy Classifiers," *in Proceedings of the 6th IEEE Joint International Information Technology and Artificial Intelligence Conference*, Chongqing, pp. 119-122, 2011.

Preecha Somwang received his MS degree in information technology from Nakhon Ratchasima College, Nakhon Ratchasima, Thailand in 2011. He is with a PhD student under faculty of information technology at Mahanakhon University of jmlo/koarea of interest includes computer network and intrusion detection.

Woraphon Lilakiatsakun received the BS degree from the King Mongkut Institute of Technology Ladkrabang, Bangkok, Thailand in 1993, the MS degree from the same university in 1998 and the PhD degree from the University of New South Wales, Australia, in 2004, all in electrical engineering. Since 2004, he has been the director of Information Technology graduate school of Mahanakorn University of Technology, Bangkok, Thailand. His recent research interest includes wireless network and internet application.