International Journal of Informatics and Communication Technology (IJ-ICT)
Vol.1, No.2, December 2012, pp. 109~118
ISSN: 2252-8776

Feature Selection and Fuzzy Decision Tree for Network Intrusion Detection

Thuzar Hlaing
University of Computer Studies, Yangon, Myanmar

Article Info

Article history:

Received Jun 06th, 2012
Revised Sept 10th, 2012
Accepted Sept 24th, 2012

Keyword:

Mutual Correlation
Fuzzy Decision Tree
True Positive
False Positive

ABSTRACT

Extra features can increase computation time, and can impact the accuracy of the Intrusion Detection System. Feature selection improves classification by searching for the subset of features, which best classify the training data. This paper proposed approach uses Mutual Correlation for feature selection which reduces from 34 continuous attributes to 10 continuous attributes. And then Fuzzy Decision Tree classifier was used for detection and diagnosis of attacks. In this paper provide good accuracy dealing with continuous attributes and prediction problem. Experimental results on the 10% KDD Cup 99 benchmark network intrusion detection dataset demonstrate that the proposed learning algorithms achieved high true positive rate (TPR) and significant reduce false positive rate (FP).

Corresponding Author:

Thuzar Hlaing
University of Computer Studies, Yangon, Myanmar
Email: thuzarhlaing.ucsy@gmail.com

1. INTRODUCTION

Intrusion detection techniques using data mining as an important application area to analyze the huge volumes of audit data and realizing performance the optimization of detection rules. The goal of feature selection is to find a feature subset maximizing some performance criterion, such as accuracy of classification. Not only that, selecting important features from input data lead to a simplification of the problem, faster and more accurate detection rates. Thus selecting important features is an important issue in intrusion detection system [1].

In 1980, the concept of intrusion detection began with Anderson's seminal paper [2]; he introduced a threat classification model that develops a security monitoring surveillance system based on detecting anomalies in user behavior. In 1986, Dr. Denning proposed several models for commercial IDS development based on statistics, Markov chains, time-series, etc [3]. In 2000, Valdes et al. [4] developed an anomaly based IDS that employed naïve Bayesian network to perform intrusion detecting on traffic bursts. In 2001, J.Gomez et al. [5] proposed a technique (genetic algorithm) to generate fuzzy rules (instead of manual design) that are able to detect anomalies.

In 2003, Kruegel et al. [6] proposed a multisensory fusion approach using Bayesian classifier for classification and suppression of false alarms that the outputs of different IDS sensors were aggregated to produce single alarm. In the same year, Shyu et al. [7] proposed an anomaly based intrusion detection scheme using principal components analysis (PCA), where PCA was applied to reduce the dimensionality of the audit data and arrive at a classifier that is a function of the principal components. In 2003, Yeung et al. [8] proposed an anomaly based intrusion detection using hidden Markov models that computes the sample likelihood of an observed sequence using the forward or backward algorithm for identifying anomalous. Dickerson et al. [9] developed the Fuzzy Intrusion Recognition Engine (FIRE) using fuzzy logic that process the network data and generate fuzzy sets for every observed feature and then the fuzzy sets are used to detect network attacks.

Huang, Pei and Goodman [10], where the general problem of GA optimized feature selection and extraction is addressed. In their paper, Huang, et al. applies a GA to optimize the feature weights of a KNN classifier and choose optimal subset of features for a Bayesian classifier and a linear regression classifier. Experiments in their paper show that the performance of all these three classifiers with feature weighing or selection by a GA is better than that of the same classifiers without a GA. They conclude that performance gain is completely dependent on what kind of classifier is used over what type of data set.

Srinivas and Sung [11] presented the use of support vector machine (SVM) to rank these extracted features, but this method needs many iterations and is very time-consuming. In the research of detection model generation, it is desirable that the detection model be explainable and have high detection rate, but the existing methods cannot achieve these two goals.

In general, IDSs can be divided into two techniques: misuse detection and anomaly detection [12], [13]. Misuse detection refers to detection of intrusions that follow well-defined intrusion patterns. It is very useful in detection known attack patterns. Anomaly detection refers to detection performed by detecting changes in the patterns of utilization or behavior of the system. It can be used to detect known and unknown attack. The anomaly detection techniques have the advantage of detecting unknown attacks over the misuse detection technique [14]. Anomaly based intrusion detection using data mining algorithms such as decision tree (DT), naïve Bayesian classifier (NB), neural network (NN), support vector machine (SVM), k-nearest neighbors (KNN), fuzzy logic model, and genetic algorithm have been widely used by researchers to improve the performance of IDS [15][16].

The rest of the paper is organized as follows: Section 2 introduces about the KDD Cup 99 Dataset. Section 3 describes data normalization and background theory. Section 4 explains a detailed description of proposed system and the experimental results in section 5.Finally, the paper is concluded with section 6.

2. KDD99 DATASET

The KDD Cup 1999 Intrusion Detection contest data KDD99 [17] is used in this experiments. This data was prepared by the 1998 DARPA Intrusion Detection Evaluation program by MIT Lincoln Labs. They acquired nine weeks of raw TCP dump data. For each TCP/IP connection, 41 various quantitative (continuous data type) and qualitative (discrete data type) features were extracted among the 41 features, 34 features are numeric and 7 features are symbolic. The data contains 22 attack types that could be classified into four main categories:

1. **Denial of Service (DOS**): In this type of attacks an attacker makes some computing or memory resources too busy or too full to handle legitimate requests, or denies legitimate users access to a machine.
2. **Remote to User (R2L):** In this type of attacks an attacker who does not have an account on a remote machine sends packets to that machine over a network and exploits some vulnerability to gain local access as a user of that machine.
3. **User to Root (U2R):** In this type of attacks an attacker starts out with access to a normal user account on the system and is able to exploit vulnerability to gain root access to the system.
4. **Probing:** In this type of attacks an attacker scans a network of computers to gather information or find known vulnerabilities. An attacker with a map of machines and services that available on a network can use this information to look for exploits.

Table 1. Different Attack Types in KDD99 Dataset

Denial of Service Attacks (DOS)	Back, Land, Neptune, Pod, Smurf, teardrop
User to Root Attacks (U2R)	Buffer_Overflow, Loadmodule, Perl, Rootkit
Remote to Local Attacks (R2L)	Ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, rwarezmaste
Probing	Satan, ipsweep, nmap, portsweep

Table 2. Input Attributes in KDD99 Dataset

No	Input Attribute	Type	No	Input Attribute	Type
1	duration	Con.	22	is_guest_login	Dis.
2	protocol_type	Dis.	23	count	Con.
3	service	Dis.	24	srv_count	Con.
4	flag	Dis.	25	serror_rate	Con.
5	src_bytes	Con.	26	srv_serror_rate	Con.
6	dst_bytes	Con.	27	rerror_rate	Con.
7	land	Dis.	28	srv_rerror_rate	Con.
8	wrong_fragment	Con.	29	same_srv_rate	Con.
9	urgent	Con.	30	diff_srv_rate	Con.
10	hot	Con.	31	srv_diff_host_rate	Con.
11	num_failed_logins	Con.	32	dst_host_count	Con.
12	logged_in	Dis.	33	dst_host_srv_count	Con.
13	num_compromised	Con.	34	dst_host_same_srv_rate	Con.
14	root_shell	Con.	35	dst_host_diff_srv_rate	Con.
15	su_attempted	Con.	36	dst_host_same_src_port_rate	Con.
16	num_root	Con.	37	dst_host_srv_diff_host_rate	Con.
17	num_file_creations	Con.	38	dst_host_serror_rate	Con.
18	num_shells	Con.	39	dst_host_srv_serror_rate	Con.
19	num_access_files	Con.	40	dst_host_rerror_rate	Con.
20	num_outbound_cmds	Con.	41	dst_host_srv_rerror_rate	Con.
21	is_hot_login	Dis.	-	-	-

3. BACKGROUND THEORY

The increase of data size in terms of number of instances and number of features becomes a great challenge for the feature selection algorithms.

3.1 Mutual Correlation

Correlation is a well known similarity measure between two random variables [18]. If two random variables are linearly dependent, then their correlation coefficient is close to ±1. If the variables are uncorrelated the correlation coefficient is 0. The correlation coefficient is invariant to scaling and translation. Hence two features with different variances may have same value of this measure. The p-dimensional feature vectors X_i of N number of instances is given by:

$$X_i = [\ {}^{i}x_1, \ldots,\ {}^{i}x_p] \qquad \text{i=1,…..., N}$$

The mutual correlation for a feature pair x_i and x_j is defined as

$$r_{x_i,x_j} = \frac{\sum_k {}^{k}x_i\ {}^{k}x_j - N\,\overline{x_i}\,\overline{x_j}}{\sqrt{(\sum_k {}^{k}x_i^2 - N\overline{x_i}^{-2})(\sum_k {}^{k}x_j^2 - N\overline{x_i}^{-2})}} \qquad (1)$$

Where k=1,….., N

If two features x_i and x_j are independent then they are also uncorrelated, i.e. r_{x_i,x_j}=0. Let us evaluate all mutual correlations for all feature pairs and compute the average absolute mutual correlation of a feature over δ features

$$r_{j,\delta} = \frac{1}{\delta}\sum_{i=1,i\neq j}^{\delta} \left|r_{x_i,x_j}\right| \qquad (2)$$

The feature which has the largest average mutual correlation

$$\alpha = \arg\max_j r_{j,\delta} \qquad (3)$$

will be removed during each iteration of the feature selection algorithm. When feature x_α is removed from the feature set, it is also discarded from the remaining average correlation, i.e.

$$r_{j,\delta-1} = \frac{\delta r_{j,\delta} - \left|r_{x_\alpha, x_j}\right|}{\delta - 1} \quad (4)$$

Algorithm 1: Feature Selection based on mutual correlation
Input: Original features set X of size N x p
Output: Reduced feature set of size N x D (D<<p)
Method:

1. Initialize δ =p
2. Discard features $X\alpha$ for α determined by Equation (3)
3. Decrement δ= δ-1, if δ<D return the Resulting D dimensional feature set and stop otherwise.
4. Recalculate the average correlations by using Equation (4).
5. Go to step 2.

3.2 Data Normalization

The original 10% KDD Cup data set where each numerical value in the data set is normalized between 0.0 and 1.0 according to the following equation:

$$x = \frac{x - MIN}{MAX - MIN} \quad (5)$$

Where,

x is the numerical value, MIN is the minimum value for the attribute that x belongs to and MAX is the maximum value.

3.3 Fuzzy Logic and C4.5 Decision Tree for Intrusion Detection

3.3.1 Fuzzy Logic

Fuzzy logic was introduced by Dr. Lotfi Zadeh of UC/ Berkeley in the 1960's as a means to model the uncertainty of natural language [19]. The normal and the abnormal behaviors in networked computers are hard to predict as the boundaries cannot be well defined. For several reasons, fuzzy logic is very appropriate for using on intrusion detection .One reason is that usually there is no clear boundary between normal and anomaly events. The use of fuzziness of fuzzy logic helps to smooth the abrupt separation of normality and abnormality. Another reason is that when to raise an alarm is fuzzy.

The KDD Cup 99 intrusion detection data set has 34 numeric attributes. Fuzzification is a process of fuzzifying numerical numbers into linguistic terms, which is often used to reduce information overload in human decision making process [20]. In this paper, triangular membership functions are used to represent fuzzy sets because of its simplicity, easy comprehension, and computational efficiency. Membership functions are usually predefined by experienced experts. The triangular membership function is denoted as $\mu_A(x)$ and is defined as:

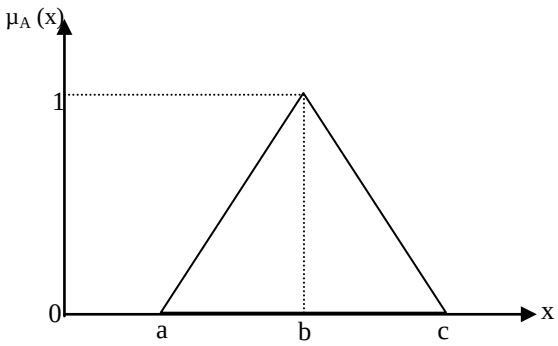

Figure 1. Fuzzy space with five fuzzy sets

$$\mu_A(x) = \begin{cases} 0 & if\ x \le a \\ \dfrac{x-a}{b-a} & if a \le x < b \\ \dfrac{c-x}{c-b} & if\ b \le x < c \\ 0 & if x \ge c \end{cases}$$

a, b and c represent the x coordinates of the three vertices of $\mu_A(x)$ in a fuzzy set A. In this paper, five fuzzy membership values (Low, Medium Low, Medium, Medium High, and High) are produced for each course score according to the predefined membership functions.

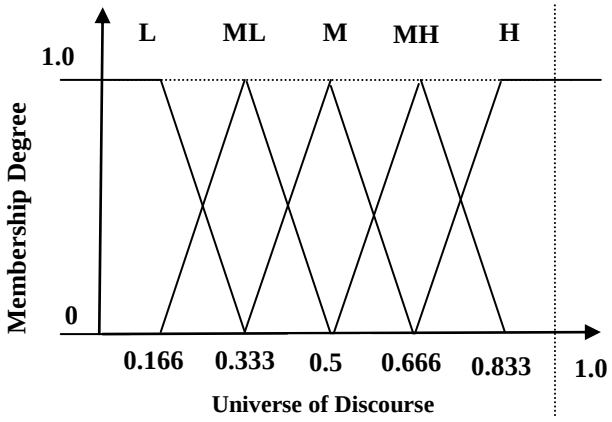

Figure 1. Fuzzy space with five fuzzy sets

3.3.2 C4.5 Decision Tree

Decision tree induction has been widely used in extracting knowledge from feature-based examples for classification and decision making. C4.5 is the algorithm proposed by R. Quinlanin 1993 [21] for building a decision tree. The C4.5 decision tree divides data items into subsets, based on attribute. If an attribute maximizes the gain ratio when dividing data into categories, it is considered useful for producing a decision tree.

3.4 Fuzzy C4.5 Decision Tree

This paper demonstrates the use of fuzzy logic to generate decision tree to classify the (continuous data).Further, the fuzzy decision tree is then converted to fuzzy rules. The fuzzy decision tree induction method used is based on minimizing the measure of classification ambiguity for different attributes. These models overcome the sharp boundary problems and good accuracy dealing with continuous attributes and prediction problems.

begin
1. Start with examples set of entry, having the weights of the examples (in root node) equalto1.
2. At any node N still to be expanded, compute the number of examples of each class. The examples are distributed in part or in whole by branches. The distributed amount of each example to a branch is obtained as the product of its current weight and the membership degree to the node.
3. Compute the standard information content.
4. At each node search the set of remaining attributes to split the node.
    4.1. Select with any criteria, the candidate attributes set to split the node.
    4.2. Compute the standard information content to each child node obtained from each candidate attribute.
    4.3. Select the candidate attribute such that information gain is maximal.
5. Divide N in sub-nodes according to possible outputs of the attribute selected in the previous step.
6. Repeat steps 2-5 to stop criteria are satisfied in all nodes.
end.

Figure 2. Fuzzy C4.5 Algorithm

4. PROPOSED FRAMEWORK

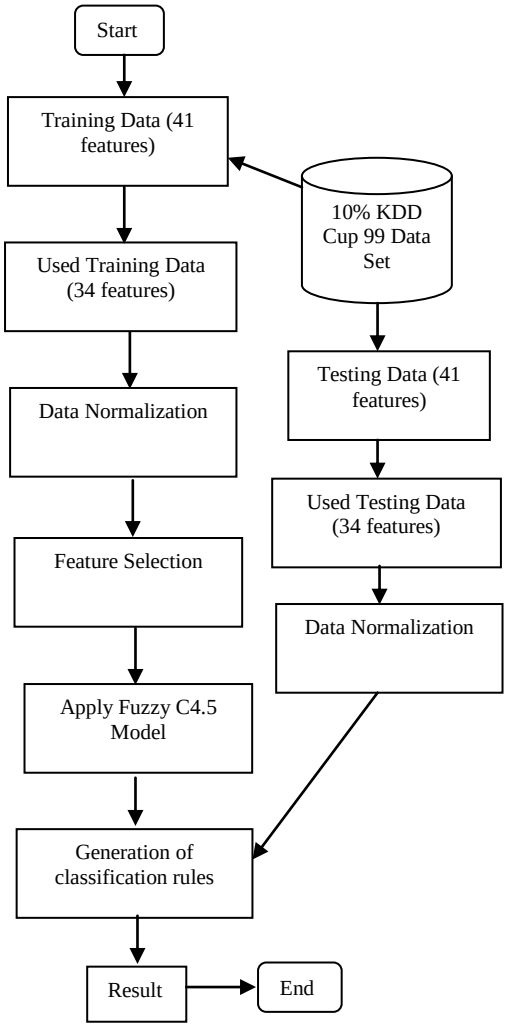

Figure 3. Overview of the Proposed Framework

Table 3. Number of correlated Features

No.	Correlated Features
A1	Duration
A2	Src_bytes
A3	Dst_bytes
A4	Wrong_fragment
A5	Urgent
A6	Hot
A7	Num_root
A8	Num_shells
A9	Srv_rerror_rate
A10	Srv_diff_host_rate

The detailed analysis of KDD data set is given in section 2. The proposed system is designed only for the continuous attributes because the major attributes in KDD data set are continuous in nature. Therefore, the proposed system have taken only the continuous attributes for instance, 34 attributes from the input dataset by removing discrete attributes. So, the proposed system is used 34 features for training data

and data normalization performs a transformation on the training data of 34 features. Then, the system is calculated optimal feature set by using Mutual Correlation (algorithm1). After calculating Mutual Correlation, the dataset is remaining 10 correlated features by experimenting. And then, the training data are analyzed by fuzzy c4.5 decision tree algorithm and the classifier is represented in the form of classification rules. Test data are used to estimate the accuracy of the classification rules.

5. Experimental Result

All experiments were performed using a 2.20GHZ Dual-Core Processor and 2GB of RAM running windows 7. In the International Knowledge Discovery and Data Mining Tools Competition only "10% KDD" dataset is employed for the purpose of training. 10% KDD training dataset consists of relatively 494021 records.

Due to the huge number of audit data records in the 10% KDD99 data set, this paper is evaluated on a subset of 10% KDD dataset by random sampling 55,285 audit records for the training phase and 35,148 records for the testing phase based on 10 correlated features.

Table 4.Training and Testing Dataset Taken for Experimentation

Attack Types	Training Dataset	Testing Dataset
Normal	25000	14863
Dos	25000	15000
U2r	52	52
R2l	1126	1126
Probing	4107	4107
Total	55,285	35,148

The proposed approach was able to generate simple classification rules. The following are some classification rules that were evolved in a sample run:

If num_root='low' and srv_rerror_rate='low' and hot='low' and srv_diff_host_rate='low' and num_shells='low' and dest_bytes='low' and src_bytes='low' and duration='low' and urgent='low' and wrong_fragment='low' Then Class is dos

If num_root='low' and srv_rerror_rate='low' and hot='low' and srv_diff_host_rate='low' and num_shells='low' and dest_bytes='low' and src_bytes='low' and duration='low' and urgent='mediumLow' Then Class is r2l

If num_root='low' and srv_rerror_rate='low' and hot='low' and srv_diff_host_rate='low' and num_shells='low' and dest_bytes='low' and src_bytes='low' and duration='medium' Then Class is normal

If num_root='low' and srv_rerror_rate='low' and hot='low' and srv_diff_host_rate='low' and num_shells='low' and dest_bytes='mediumLow' and src_bytes='high' and duration ='low' and wrong_fragment='low' and urgent='low' Then Class is u2r

If num_root='low' and srv_rerror_rate='low' and hot='low' and srv_diff_host_rate='high' and src_bytes='low' and duration='low' and dest_bytes='low' and wrong_ fragment= 'low' and urgent='low' and num_shells='mediumLow' Then Class is probing

If num_root='low' and srv_rerror_rate='high' and srv_diff_ host_rate='high' and duration='high' Then Class is probing

If num_root='mediumLow' Then Class is normal

If num_root='medium' Then Class is dos

If num_root='high' Then Class is normal

To estimate the performance of the system the following formulas are used.

True Positive Rate = TP/TP+FN

False Positive Rate= FP/FP+TN

Classification Rate= Number of classified patterns*100%
Total number of patterns

Table 5. Detailed Accuracy by Class

Class	True Positive Rate (TP)	False Positive Rate (FP)	Precision	Recall
Normal	0.999	0.004	0.996	0.999
Dos	0.999	0.01	0.988	0.999
U2r	0.5	0	0.743	0.5
R2l	0.951	0	0.992	0.951
Probing	0.92	0	0.995	0.92

Table 6 compare the different algorithm performance, the total classification accuracy of proposed algorithm is better than other algorithm.

Table 6. Different algorithms Performances

Algorithm	Accuracy
Mutual Correlation+ Fuzzy C4.5	99.1734%
Neural Network+ SVM[22]	96.5%
Fuzzy Logic[23]	94.6%
C4.5[24]	93.67%

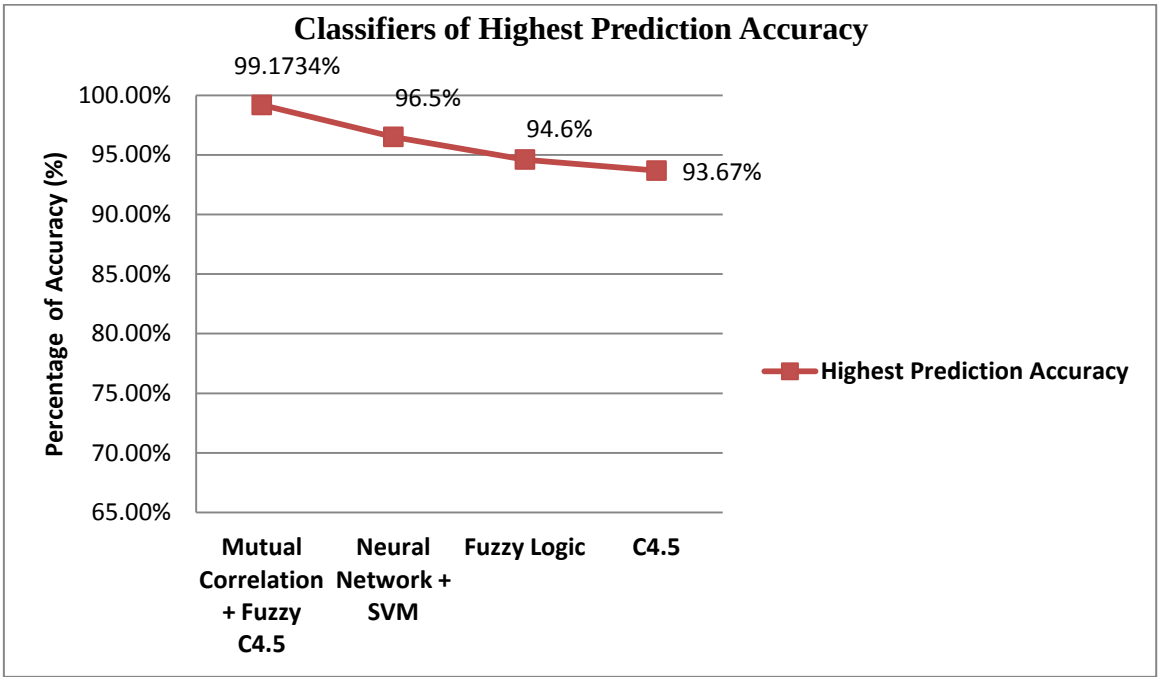

Figure 4. Classifiers with Highest Prediction Accuracy

Figure 4 shows the classifiers with highest prediction accuracy. The proposed combine methods predict better than other combine methods.

6. CONCLUSION

As security incidents become more numerous, IDS tools are becoming increasingly necessary. It is very likely that IDS capabilities will become core capabilities of network infrastructure (such as routers, bridges and switches) and operating systems. Feature selection of Mutual Correlation and fuzzy C4.5 classifier were designed to build the system more accurate for attack detection, using fuzzy logic based on numeric numbers. By analyzing the result, the overall performance of the proposed system is improved significantly and it achieved 99% accuracy.

REFERENCES

[1] B. J. Kim and I. K. Kim, "Machine Learning Approach to Real time Intrusion Detection System." In Proceedings of 18th. Australian Join Conference on Artificial Intelligence, 2005, Sydney, Australia. Vol. 3809. Pp. 153-163.
[2] James P. Anderson, "Computer security threat monitoring and surveillance," Technical Report 98-17, James P. Anderson Co., Fort Washington, Pennsylvania, USA, April 1980.
[3] Dorothy E. Denning, "An intrusion detection model," IEEE Transaction on Software Engineering, SE-13(2), 1987, pp. 222-232.
[4] A. Valdes, K. Skinner, "Adaptive model-based monitoring for cyber attack detection," in Recent Advances in Intrusion Detection Toulouse, France, 2000, pp. 80-92.
[5] J.Gomez and D.Dasgupta, "Evolving Fuzzy Classifiers for Intrusion Detection", Proceeding of the IEEE Workshop on Information Assurance, United States Military Academy, June 2001.
[6] C. Kruegel, D. Mutz, W. Robertson, F. Valeur, "Bayesian event classification for intrusion detection," in Proc. of the 19th Annual Computer Security Applications Conference, Las Vegas, NV, 2003.
[7] M.L. Shyu, S.C. Chen, K. Sarinnapakorn, L. Chang, "A novel anomaly detection scheme based on principal component classifier," in Proc. of the IEEE Foundations and New Directions of Data Mining Workshop, Melbourne, FL, USA, 2003, pp. 172-179.
[8] D. Y. Yeung, and Y. X. Ding, "Host-based intrusion detection using dynamic and static behavioral models," Pattern Recognition, 36, 2003, pp. 229-243.
[9] J.E. Dickerson, J.A. Dickerson, "Fuzzy network profiling for intrusion detection," In Proc. of the 19th International Conference of the North American Fuzzy Information Processing Society (NAFIPS), Atlanta, GA, 2000, pp. 301-306.
[10] Huang, Z., Pei, M., Goodman, E., Huang, Y., and Li, G.Genetic algorithm optimized feature transformation: a comparison with different classifiers. In Proc. GECCO 2003, pp. 2121-2133.
[11] Srinivas, M., Sung, A., "Feature Ranking and Selection for Intrusion Detection". Proceedings of the International Conference on Information and Knowledge Engineering, 2002.
[12] E. Biermann, E. CloeteandL.M.Venter,"A comparison of intrusion detection Systems", Computer and Security, vol.20, pp.676-683, 2001.
[13] T.Verwoerd and R.Hunt,"Intrusion detection techniques and approaches", Computer Communications, vol.25, pp.1356-1365, 2002.
[14] E. Lundin and E. Jonsson, "Anomaly-based intrusion detection: privacy concerns and other problems", Computer Networks, vol.34, pp.623-640, 2002.
[15] Barbara, Daniel, Couto, Julia, Jajodia, Sushil, Popyack, Leonard, Wu, and Ningning, "ADAM: Detecting intrusion by data mining," IEEE Workshop on Information Assurance and Security, West Point, New York, June 5-6, 2001.
[16] T. Shon, J. Seo, and J.Moon, "SVM approach with a genetic algorithm for network intrusion detection," In Proc. of 20th International Symposium on Computer and Information Sciences (ISCIS 2005), Berlin: Springer-Verlag, 2005, pp. 224-233.
[17] KDD Cup 1999 Dataset: http://kdd.ics.uci.edu/ databases/kddcup99
[18] Veerabhadrappa, L. Rangarajan,"Multi-level Dimensionality Reduction Methods using Feature Selection and Feature Extraction," IJAIA, Vol.1, No.4, October 2010.
[19] L.A. Zadeh, Fuzzy Sets, Information and Control, 8(3), 338-353, 1965
[20] M. R. Civanlar and H. J. Trussell, "Constructing membership functions using statistical data," Fuzzy Sets and Systems, vol. 18, 1986, pp. 1-14.
[21] J.R.Quinlan.C4.5:ProgramforMachineLearning.MorganKaufmannPublishers, 1993.
[22] S.K.Gupta, Dr.S.M.Krisha "Enhancing Technique for Intrusion Detection Using Neural Network and SVM Classifier," Global Journal of Computer Science and Technology, Vol 12, Issue 13 version 1.0, February 2012.
[23] R.Shanmugavadivu, N.Nagarajan,"An Anomaly-Based Network Intrusion Detection System Using Fuzzy Logic". International Journal of Computer Science and Information Security, Vol.8, No.8, November 2010.
[24] K.Saravanan," An Efficient Detection Mechanism for Intrusion Detection Systems Using Rule Learning Method". International Journal of Computer and Electrical Engineering, Vol.1, No.4, October, 2009.

BIOGRAPHY OF AUTHORS

I received B.C.Sc, B.C.Sc(Hons:) and M.C.Sc degrees in Computer Science from University of Computer Studies, Yangon(UCSY) in 2005,2006 and 2009 respectively. During 2007-2009, I served as a tutor in Computer Application Department of University of Computer Studies,Myeik. From 2009 to 2010, I attened PH.D coursework in UCSY. Now, I'm doing my research which is related with Data Mining.