http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 13 (1): 133-139, 2014
ISSN 1812-5638 / DOI: 10.3923/itj.2014.133.139

Unsupervised Intrusion Detection based on FCM and Vote Mechanism

[1]Longlong Li, [1]Qin Chen, [2]Shuiming Chi and [1]Xiaohang Liu
[1]College of Computer Science, Hangzhou Dianzi University, 310018, Hangzhou, China
[2]Jinhua Biqi Network Technology Co., Ltd., 321017, Jin Hua, China

Abstract: Two different Intrusion Detection Systems (IDS) are by far dominant in current commercial network attacks detection: misuse detection and anomaly detection. Both of them are motivated by the limitations of knowledge-based approaches which may cause the limitations of detection performance. In this paper, we present an Unsupervised Intrusion Detection System (UIDS) capable of detecting network attacks based on FCM-Vote algorithm which do not need any previous knowledge of either the characteristics of attack or training set. FCM-Vote based IDS uses a novel network malicious ows detection approach based on FCM clustering and clustering fusion. The new approach improve the robustness of UIDS which based on clustering and has high accuracy in detecting the network malicious ows. In order to validate these advantages of such an Unsupervised Intrusion Detection System, we evaluate its ability in a well-known existing dataset of traffic features. We particularly show the low rate of false alarm of FCM-Vote based IDS by comparing its performance against traditional anomaly detection which based on limited knowledge.

Key words: Intrusion detection, clustering algorithm, clustering of fusion, subspace clustering

INTRODUCTION

Distributed Denial of Service attacks (DDoS), buffer overflow attacks, network scanning activities, and spreading worms or viruses are examples of the different threats that daily compromise the integrity and normal operation of the network.Intrusion detection is a technique to discover invasive by collecting and analyzing information on the protected systems. Its main function is to provide real-time monitoring, detection and recognition system intrusion or attempted intrusion alarm for network and computer systems (Hansman and Hunt, 2005). Two different Intrusion Detection Systems (IDS) are in current commercial network attacks detection: Misuse detection and anomaly detection. Both of them depend on the limitations of knowledge-based approaches. But it is not easy to maintain an accurate and up-to-date normal-operation profile, particularly in a dynamic and evolving context where new services and applications are constantly emerging (Casas *et al*., 2012). Unsupervised Anomaly Detection (UAD) has been studied in the last few years which uses Data Mining techniques to extract patterns and similar structures, without relying on signatures or baseline traffic profiles (Androulidakis *et al*., 2009). Some methods for unsupervised detection of network attacks have been proposed in the past (Portnoy *et al*., 2001; Eskin *et al*., 2002; Leung and Leckie, 2005; Lakhina *et al*., 2005) the majority of them are based on clustering techniques and outliers detection. clustering addresses multiple-class problems without modifying the characteristics of the analyzed traffic. Unfortunately, even if hundreds of clustering algorithms exist (Jain, 2010), one of the major drawbacks in current cluster analysis techniques is the lack of robustness.

In this study, we present an approach of UAD based on FCM and Voting mechanism which places the emphasis on the robustness of clustering results and the capable of detecting malicious ows without any previous knowledge. The main idea is to combine the notions of Sub-Space Clustering (SSC) (Parsons *et al*., 2004) and the voting mechanism to produce these multiple independent partitions. We divide the process of intrusion detection into four phases: (1) Traffic detection, (2) Features clustering analysis and (3) Outliers analysis. The structure of UAD system is shown in Fig. 1.

The first step is just to detect the presence of an anomalous time slot where an attack or other kind of traffic anomaly might be hidden.We simply compute the difference between consecutive time slots of some very basic traffic metrics such as number of bytes, packets and IP ows. An abrupt change in any of metrics ags the presence of an anomalous behavior which triggers the second step of the analysis. Modeling anomalies as abrupt changes in network traffic is a standard approach (Thottan and Ji, 2003; Barford *et al*., 2002). In the second step, we extract the data flow characteristics from the previous step. Before we take the process of cluster

Corresponding Author: Longlong Li, College of Computer Science, Hangzhou Dianzi University, 310018, Hangzhou, China

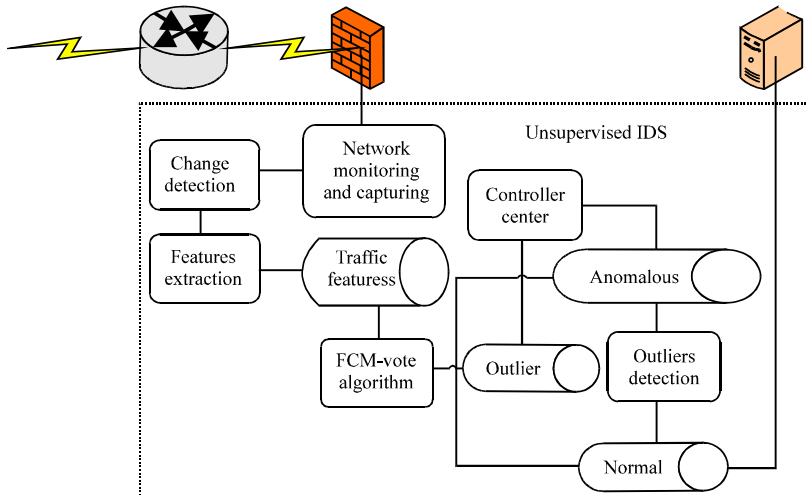

Fig. 1: Structure of UAD system

analysis, we should reduce the dimension of the traffic which may divide the high-dimensional data into several sub-spaces. Clustering results are usually divided into three categories: The normal ows, anomaly ows and outliers. Then we submit the data which have been identified as anomaly ows to control center. Finally, we analyze the outliers which generated by the second step. We take process of outliers analysis by voting mechanism to calculate outlier weight of each outlier ow. The outlier ow, whose weight exceeds the noise threshold value, will be defined as an anomaly ow and be submit to he control center. Experimental results show that the FCM-Vote based UAD system can detect malicious automatically and efficiently.

The remainder of this study is organized as follows: Section 2 will describe some applications on intrusion detection which based on clustering algorithm. Following that, We conduct a description of the current challenges and present our solution ideas based on the actual situations in section 3. Section 4 demonstrates the design of the core algorithm solutions in the following section and make a detailed description of the algorithm structure and function of the main module. Section 5 evaluates the performance in different aspects, by the comparing with other intrusion detection approaches to show the advantages in reducing the false positive rate in the authentication algorithm and increasing detection efficiency. Finally, we summarize this study and look into the future trends in section 6.

RELATED WORK AND CONTRIBUTION

Two different Intrusion Detection Systems (IDS) are by far dominant in current commercial network attacks detection: Misuse detection and anomaly detection. Anomaly intrusion detection system means the establishment of the normal pattern outline, if the difference value between the our profile and the normal value exceeds the specified threshold, the system will report an incident. Aim to build the target of detection by intrusion as the destination, anomaly intrusion detection has the advantage that does not rely on signatures. However, how to build targets used to detects abnormal or define the outline of the normal mode is difficult to resolve.

The research on IDS developed rapidly in the past few years. Some approaches of unsupervised detection have been proposed (Portnoy *et al*., 2001; KDD Cup, 1999; Eskin *et al*., 2002). The main idea of these methods is based on the clustering. Portnoy *et al*. (2001) uses a single-linkage hierarchical clustering method to cluster data from the KDD99 (KDD Cup, 1999) dataset, based on the standard Euclidean distance for inter-patterns similarity. Eskin *et al*. (2002) reports improved results in the same data-set, using three different clustering algorithm: Fixed-Width clustering, an optimized version of k-NN and one class SVM. Leung and Leckie (2005) presents a combined density-grid-based clustering algorithm to improve computational complexity, obtaining similar detection results. In recent years, studies and experiments show that Clustering Fusion can improve robustness, stability and other properties of intrusion detection. Topchy (Guha *et al*., 2000) found that Clustering Fusion algorithm has a better clustering effect than a single clustering algorithm. This study will introduce a clustering analysis based on Data Mining and the Clustering Fusion, we combine these technologies and implement the traffic clustering in our FCM-vote based IDS.

FRAMEWORK OF RESOLUTION

Traffic detection: We capture ows in consecutive time slots of fixed length ΔT and analyze the different time-series associated with each aggregation key at each new time slot. An abrupt change which may present an anomalous behavior will trigger the step of the analysis. We model the abrupt changes in network traffic by a standard approach.

Features clustering analysis

Sub-space clustering (SSC): Network traffic is often massive and high-dimensional, for example, the KDD99 data sets used in this study includes 41 features. This fact makes the final clustering results become very unstable because of the interference from irrelevant attributes. It is intolerable to the security workers who are focus on accuracy of the network intrusion detection. Furthermore, the detection of the high dimensional ows increases the time cost.

Therefore, in order to reduce the dimension of aggregated flows by Sub-Space Clustering (SSC), we need to divide subspaces for characteristics based on correlation properties of the network traffic before performing the process of clustering analysis. Without loss of generality, let $S = \{S_1, S_2,..., S_n\}$ be the set of n aggregated-flows. Each flow $S_i = \{A_{i1}, A_{i2},..., A_{im}\}$ is described by a vector of m features, e.g., src_bytes, dst_bytes, srv_count and so on. $A_i = \{A_{i1}, A_{i2},..., A_{im}\}$ is the corresponding vector of S_i and $A_i = \{A_1, A_2,..., A_n\}$ will be referred to as the feature space. Through attribute subspace partition, each subspace of the attributes set is independent which is perfectly adapted for parallel computing architectures. In this study, we show the dimensionality reduction of aggregated flows in KDD99 data set and separate the total 41 features of KDD99 data set into four categories: Features of TCP connection, features of TCP content, features of time-based network traffic and features of host-based network traffic, in which the features of time-based network traffic statistics contains "same host" features and "same service" features. Each category is constructed using only r traffic features; this permits us to analyze the structure of high-dimension aggregated flows using a finer-grained resolution. We refer the categories constructed according to certain rules as the subspace of the total feature space. Instead of directly partitioning the complete feature space, clustering analysis will be carried out within each subspace. We separate the complete feature space into k subspaces and let the set of subspaces $P = \{P_1, P_2,..., P_k\}$ and $P_j = \{S^j_1, S^j_2,..., S^j_n\}$ is one of the subspaces. Each S^j_i in the subspace P_j is constructed using only $r \le m$ traffic features. We cluster the flows which are described by a set of features in different subspaces of smaller dimensions and obtain different partitions of the flows in S and separate each subspace P_j into several partitions by clustering analysis.

Clustering analysis: Clustering is one of the critical methods in the field of Data Mining. Clustering algorithm is to separate the curtain subspace P_j into the c partitions and $V_j = \{V_{j1}, V_{j2},..., V_{jc}\}$ is the set of c clustering centers in sub-space P_j. $V_{ja}(1 \le a \le c)$ is one of the clustering centers. We assume that $S^j_i = \{A_{i1_1}, A_{i1_2},..., A_{i1_r}\}$ is one of the feature vectors in subspace P_j and $A_{i1_1}, A_{i1_2},..., A_{i1_r}$ are r traffic features in subspace P_j. In this study, we introduce the idea of Fuzzy C-Means (FCM) algorithm into our process of clustering and let the matrix U_j be the membership matrix of subspace P_j where η is the membership threshold. We set the c = 2 and define the element $u_j[i][a]$ of U_j as the membership of the feature vector S^j_i belonging to the clustering centers V_{ia}, where:

$$u_j[i][a] \in [0,1], \sum_{a=1}^{2} u_j[i][a] = 1$$

As we start the process of clustering, we need to initialize the initial clustering centers $V^0_j = \{V^0_{j1}, V^0_{j1}\}$. The clustering algorithm also need to detect the outliers, we define the set O_j to save the outliers in subspace P_j. FCM-based clustering analysis shown in Algorithm 1 presents a pseudo-code for Clustering Analysis. There are several input parameters in this algorithm: Subspace P_j, the number of clusters c, fuzzy factor M and the iteration termination condition σ. The objective function of clustering algorithm can be expressed as:

$$J(U,V) = \sum_{i=1}^{n}\sum_{a=1}^{2}(u_j[i][a])^M d_{ia}^2 \quad (1)$$

$d_{ia} = \|S^j_i - V_{ja}\|$ is the Euclidean distance between the center V_{ja} and the feature vector S^j_i, M is a fuzzy weighting parameter. FCM algorithm is to find the minimum value of the objective function J by Eq. 2:

$$\sum_{a=1}^{2} u_j[i][a] = 1, u_j[i][a] \in [0,1] \quad (2)$$

In order to make the objective function J(U, V) of Eq. 1 to be the minimum value, center V_{ja} and membership matrix U_j can be calculated by Eq. 3:

$$V_{ja} = \frac{\sum_{i=1}^{n}(u_j[i][a])^M S^j_i}{\sum_{i=1}^{n}(u_j[i][a])^M} \quad (3)$$

If $d_{ia} \neq 0$:

$$u_j[i][a] = \left(\sum_{g=1}^{2}\left(\frac{\|S^j_i - V_{ja}\|}{\|S^j_i - V_{jg}\|}\right)^{\frac{2}{M-1}}\right)^{-1} \quad (4)$$

If $d_{ia} = 0$:

$$u_j[i][a] = 1,\ u_j[g][a] = 0,\ g \neq i \tag{5}$$

Algorithm 1: FCM-based clustering analysis

Require:
Subspace, P_j;
The number of clusters, c;
Fuzzy factor, M;
Membership threshold, η;
The iteration termination condition, σ;
Ensure:
The current subspace clustering center set, V_j;
The membership matrix, U_j;
The outlier set, O_j.
1: Initialize V^0_j;
2: check the value of σ;
3: save the size of as n;
4: **for** i in 1:n do
5: $u_j[i][a] = \left(\sum_{g=1}^{2} \left(\frac{\left\| S_i^j - V_{ja}^0 \right\|}{\left\| S_i^j - V_{jg}^0 \right\|} \right)^{\frac{2}{M-1}} \right)^{-1}$
6: **end** for
7: **for** a in 1:2 do
8: $V_{ja} = \frac{\sum_{i=1}^{n} (u_j[i][a])^M S_i^j}{\sum_{i=1}^{n} (u_j[i][a])^M}$
9: if $\|V_{ja}-V^0_{ja}\| \leq \sigma$ then
10: **break**
11: **else then**
12: $V^0_{ja}-V_{ja}$;
13: **end if**
14: **end for**
15: **for** a in 1:2 **do**
16: $V_j = V_j \cup V_{ja}$
17: **for** i in 1:n **do**
18: **save** $u_j[i][a]$ to set U_j
19: **end for**
20: **end for**
21: **for** i in 1:n **do**
22: **if** $u_j[i][1] \leq \eta$ and $u_j[i][2] \leq \eta$ **then**
23: $O_j = O_j \cup S_i^j$
24: **end if**
25: **end for**
26: **return** (V_j, U_j, O_j);

Line 1-2 is responsible for initializing the cluster center as V^0_{ja} and check the iteration criteria $\sigma>0$. Then we save the number of feature vectors in the sub-space into the variable n. We traverse the subspace and calculate the membership $u_j[i][a]$ by Eq. 4 in line 4-6. After obtaining the membership of each feature vector to the cluster center, we calculate the changed cluster center as V_{ja} through Eq. 3 in line 7-14 and end the traverse when $\|V_{ja}-V^0_{ja}\| \leq \sigma$, the traversal continues if $V^0_{ja} = V_{ja}$. We use the set V_j and matrix U_j in line 15-20, respectively, to save the current subspace clustering center into the set V_j and set the U_j to be the membership matrix. Line 21-25 shows the process of outlier analysis. we conduct outlier analysis by using the determine the membership threshold η and calculate the membership of current cluster center. If the membership $u_j[i][a](1 \leq a \leq 2)$ is greater than η, we include S^j_i into the clustering result which contains the current clustering center V_{ja}. Or we continue to calculate the membership of the next cluster center. If there is a feature vector cannot be include into any clustering result, we will judge it as outlier and save the outlier feature vectors into the set O_j. Then we complete the process of fuzzy clustering. Finally, we output the result (V_j, U_j, O_j). Each subspace will have two clustering centers usually. We treat flows belong to the big-size cluster result as the normal flows and the ones belong to the small-size as the abnormal flows. The outlier flows will be ones which do not belong to both of them.

Clustering fusion based on vote mechanism: In order to obtain the results of the detection, we merge the clustering centers $V = \{V_1, V_2,..., V_k\}$ where $V_j = \{V_{j1}, V_{j2}\}$, $1 \leq j \leq k$. We design a consensus function based on vote mechanism which combines the clustering results through the co-associated matrix and generates the final results $R = \{r_1, r_2\}$. We define c = 2 as the number of the clustering center and let $r_a(1 \leq a \leq 2)$ be the set of the final results. Let $U = \{U_1, U_2,..., U_k\}$ be the set membership matrix of k subspaces where U_j is the membership matrix of subspace P_j. The Vote based Clustering Fusion algorithm selects $V_j = \{V_{j1}, V_{j2}\}$ as a category identifier to identify the category of other classes by the largest number of votes. Clustering results are fused by designing the consensus vote mechanism clustering function. The algorithm of clustering fusion based on vote mechanism can be described as algorithm 2.

Algorithm 2: Clustering fusion based vote mechanism

Require:
The results of subspace clustering of feature vectors, $V = \{V_1, V_2,..., V_k\}$, $V_j = \{V_{j1}, V_{j2}\}$, $1 \leq j \leq k$;
Subspaces of feather space, $P = P_1, P_2,..., P_k$, $P_j(1 \leq j \leq k)$;
Set of membership matrix, $U = \{U_1, U_2,..., U_k\}$;
Membership threshold, η
Ensure:
The final results $R = \{r_1, r_2\}$
1: Initialize matrix B[n][2]
2: Select the first clustering result V_1 as a reference to switch the category label
3: **for** i in 1:n **do**
4: **for** j in 1:k **do**
5: **for** a in 1:2 **do**
6: **if** $\eta \leq u_j[i][a]$ **then**
7: B[i][a]++
8: **end if**
9: **end for**
10: **end for**
11: **if** $B[i][2] \leq B[i][1]$ **then**
12: $r_1 = r_1 \cup S_i$
13: **else**
14: $r_2 = r_2 \cup S_i$
15: **end if**
16: **end for**
17: **return** $R = \{r_1, r_1\}$;

We initialize matrix B[n] (Casas *et al*., 2012) to save the votes which belong to a certain category in line 1. Line 2 shows the process of selecting the first clustering results as a reference to switch the category label. Then, we traverse subspaces and select a clustering result as a benchmark and determine which category belongs to by the largest number of votes. Finally, we'll return $R = \{r_1, r_1\}$ as the set of fusion results.

Attack detection: Finally, the results of clustering fusion will be two clustering center, respectively implement normal traffic clustering and anomaly traffic clustering. The flows in anomaly traffic clustering can be considered as the malicious flows that may contain network attack and the our FCM-Vote based detection will filter them to ensure security of current network. Then we take the process of outlier analysis.

In the case of outlier analysis, we build a rareness or dissimilarity ranking of flows. Each value of this dissimilarity value corresponds to the cumulative separation of an outlier to the biggest cluster in each of the different subspaces. The biggest cluster in each subspace statistically represents normal operation traffic; hence, those flows that present a bigger dissimilarity value correspond to flows that are very different from the majority of the traffic through the different subspaces. The flows with biggest similarity value with the center of small-size clusters are flagged as anomalies. Since, each subspace in the clustering process may produce outlier, we also need to merge the results of outlier analysis by vote mechanism. We identify the outlier flows by calculating the maximum number of outlier votes. We also need to set the outlier threshold λ. If the number of outlier votes is greater than λ, the current feature vector will be regarded as anomalies. The process of vote mechanism will refer to algorithm 2.

EVALUATION

Simulation settings: To validate the proposed FCM-Vote based Intrusion Detection System, we will use KDDCUP99 intrusion detection data sets as experimental subjects. Therefore, in order to avoid dependence on measurement selection and eliminate the impact to the clustering results caused by the differences among the attribute measures, we need to pretreat attribute values. The pretreatment can be divided into two steps: Standardization and normalization. The process of normalization will normalize the standardization value to the interval [0,1].

Setting of M: Before carrying out the specific intrusion detection test, we need to test the weight M which is used to determine the degree of clustering fuzzy. M is usually taken [1.25, 2.5] in practical applications and (O'Callaghan *et al*., 2002) proposed the M-optimal selection method based on fuzzy decision. We test the value of preferably M in the intrusion detection system 2000 times and calculate the variance of clustering results in Fig. 2.

Figure 2 shows the variance of the value of M in the selected range. When M take the value of 2, the clustering results can be the most ideal, so FCM algorithm will take 2 as the preferred value of M in our experiment.

Clustering results: To show the effect of clustering, we take 10 percent of KDD99 as the subjects and cluster in one of the subspaces, the experimental results are shown in Fig. 3.

Figure 3 presents the detection and automatic characterization of the anomaly traffic which may contain network attacks such as DDoS attack, buffer over flow attacks, network scanning activities and so on in one of the subspaces separated by SSC. We define a curtain

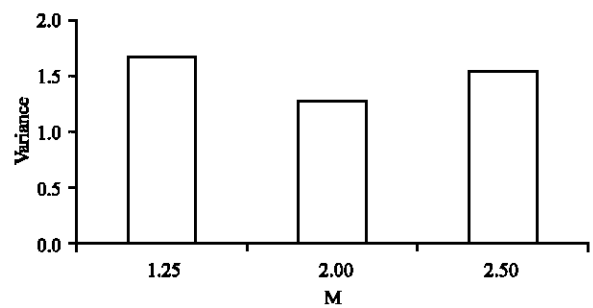

Fig. 2: Variance of clustering results by different value of M

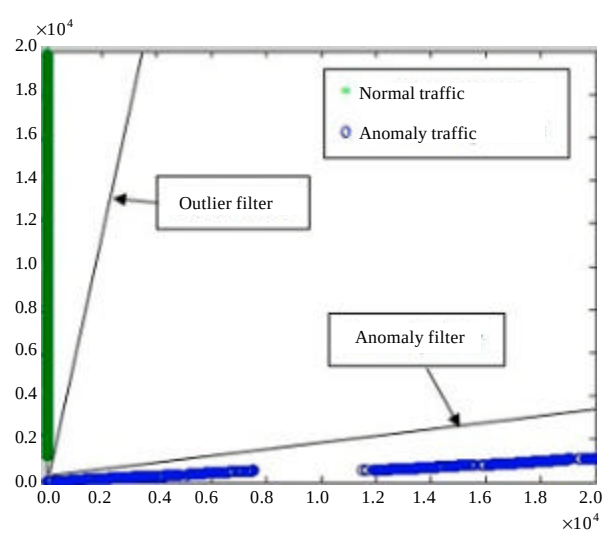

Fig. 3: Clustering results of FCM algorithm c

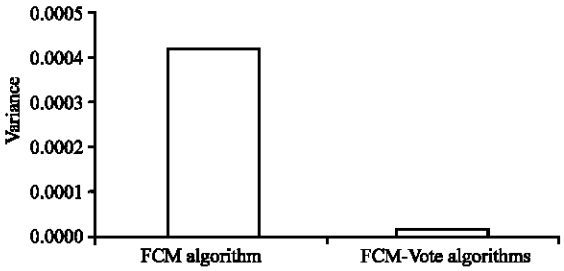

Fig. 4: Comparison on the variance of clustering results

threshold in the outlier filter and anomaly filter and compare the Euclidean distance between feature vector and the clustering center with the threshold to determine the category of the current feature vectors. We demonstrate the outstanding performance of FCM clustering in Fig. 3. Ordinate represents the Euclidean distance between feature vector and the small-size clustering center. On the other hand, Abscissa represents the Euclidean distance between feature vector and the big-size clustering center. We filter the anomaly flows by anomaly filter which identified the small-size clustering results and filter the outlier flows by comparing the Euclidean distance between flow and the big-size clustering center with a curtain threshold.

Intrusion detection results: To show the performance of our FCM-Vote based IDS, we test 20,000 data and gradually increase the number of data to 100000, Table 1 shows the results of the experiment.

Our FCM-Vote based IDS improve the performance of the existing anomaly detection by using the Data Mining techniques in traffic clustering that does not rely on previous knowledge. Table 1 shows that our system maintaining a high accuracy.

Robustness of clustering fusion: We compared the variance of clustering results after running 20,000 randomly selected set of data through FCM-Vote algorithms and single FCM algorithm 100 times in Fig. 4.

Figure 4 shows that the results of comparing the results obtained by different clustering strategies. The fact is that clustering fusion algorithm based on vote mechanism has higher robustness than the single clustering algorithm.

Rate of false alarm: We take 20000 samples from Table 1 as the training set of supervised anomaly detection. Then we compare the rate of false alarm between FCM-Vote based detection and traditional supervised anomaly detection on the random data set sample from KDD99. The experimental results are demonstrated in Fig. 5.

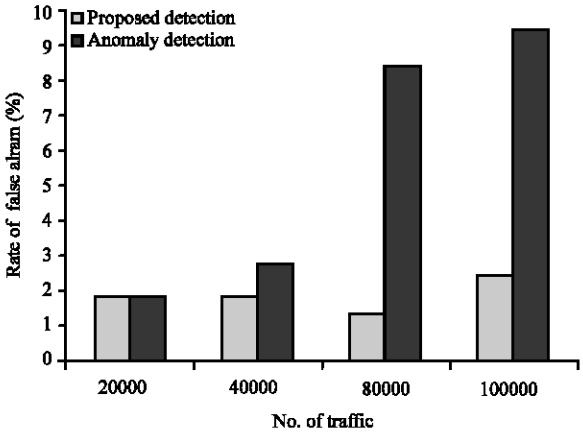

Fig. 5: Comparison on the rate of false alarm

Table 1: Test results of intrusion detection

Data	Missing rate (%)	False rate (%)	Accuracy (%)
20000	0.73	1.885	97.385
40000	0.81	1.786	97.404
80000	0.93	1.362	97.708
100000	1.23	2.428	96.342

The results of Fig. 5 reveal the fact that the proposed IDS maintains a lower false alarm rate in intrusion detection than the traditional supervised anomaly detection dose. As the traditional anomaly detection relies on the training set, the imperfection of the training set which selected by developer may cause the high rate of false alarm in the case of the limited training sets.

CONCLUSION

We proposed a novel intrusion detection approach based on Sub-Space Clustering (SSC) and clustering fusion in this study and implement the new approach in our FCM-Vote based Unsupervised Intrusion System (UIDS) to implement the detection of network attack hidden in the daily traffic and verify the advantages in accuracy, efficiency and robustness of new approach by comparing with the traditional methods of intrusion detection in our experiment. In a diametrically oppo-site perspective, we place the emphasis on the development of unsupervised knowledge-independent detection algorithms. There are many valuable issues to be studied, such as collaborative intrusion detection system, attack intention recognition algorithms and real-time response of intrusion would be direction of development in the future.

ACKNOWLEDGMENTS

This study is supported in part by the Zhejiang Provincial Technical Plan Project (No. 2012C11026-4).

REFERENCES

Androulidakis, G., V. Chatzigiannakis and S. Papavassiliou, 2009. Network anomaly detection and classification via opportunistic sampling. IEEE Network, 23: 6-12.

Barford, P., J. Kline, D. Plonka and A. Ron, 2002. A signal analysis of network traffic anomalies. Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement, November 6-8, 2002, Marseille, France, pp: 71-82.

Casas, P., J. Mazel and P. Owezarski, 2012. Knowledge-Independent traffic monitoring: Unsupervised detection of network attacks. IEEE Network, 26: 13-21.

Eskin, E., A. Arnold, M. Preraua, L. Portnoy and S.J. Stolfo, 2002. A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data. Data Mining for Security Applications. Kluwer Academic Publishers, Boston.

Guha, S., N. Mishra, R. Motwani and L. O'Callaghan, 2000. Clustering data Streams. Proceedings of the 141st Annual Symposium on Foundations of Computer Science, November 12-November 14, 2000, Redondo Beach, California, pp: 1359-366.

Hansman, S. and R. Hunt, 2005. A taxonomy of network and computer attacks. Comput. Secur., 24: 31-43.

Jain, A.K., 2010. Data clustering: 50 years beyond K-means. Pattern Recogn. Lett., 31: 651-666.

KDD Cup, 1999. Data. Online. http://kdd.ics.uci.edu/

Lakhina, A., M. Crovella and C. Diot, 2005. Mining anomalies using traffic feature distributions. Proceedings of the Conference on Applications, Technologies, Architectures and Protocols for Computer Communications, Volume 35, October 2005, ACM New York, pp: 217-228.

Leung, K. and C. Leckie, 2005. Unsupervised anomaly detection in network intrusion detection using clusters. Proceedings of the 28th Australasian Conference on Computer Science, January 2005, Newcastle, Australia, pp: 333-342.

O'Callaghan, L., N. Mishra, A. Meyerson, S. Guha and R. Motwani, 2002. Streaming-data algorithms for high-quality clustering. Proceedings of the 18th International Conference on Data Engineering, February 26-March 1, 2002, San Jose, CA., USA., pp: 685-694.

Parsons, L., E. Haque and H. Liu, 2004. Subspace clustering for high dimensional data: A review. Sigkdd Explorat., 6: 90-105.

Portnoy, L., E. Eskin and S. Stolfo, 2001. Intrusion detection with unlabeled data using clustering. Proceedings of ACM CSS Workshop on Data Mining Applied to Security, November 5-8, 2001, Philadelphia, PA., USA.

Thottan, M. and C. Ji, 2003. Anomaly detection in IP networks. IEEE Trans. Signal Proc., 51: 2191-2204.