International Journal of Engineering Research and Development
eISSN : 2278-067X, pISSN : 2278-800X, www.ijerd.com
Volume 2, Issue 2 (July 2012), PP. 44-48

Implementation of Fuzzy *c*-Means and Outlier Detection for Intrusion Detection with KDD Cup 1999 Data Set

S. Songma[1], W. Chimphlee[2], K. Maichalernnukul[3], P. Sanguansat[4]
[1,3]*Faculty of Information Technology, Rangsit University, Muang-Ake, Phathumthani 12000, Thailand*
[2]*Faculty of Science and Technology, Suan Dusit Rajabhat University, Bangkok 10300, Thailand*
[4]*Faculty of Engineering and Technology, Panyapiwat Institute of Management, Nonthaburi 10200, Thailand*

***Abstract*— In this paper, a two-phase method for computer network intrusion detection is proposed. In the first phase, a set of patterns (data) are clustered by the fuzzy *c*-means algorithm. In the second phase, outliers are constructed by a distance-based technique and a class label is assigned to each pattern. The KDD Cup 1999 data set is used for the experiment. The results show that, for binary classification (i.e., normal or attack), the proposed method achieves a higher detection rate and a greater overall accuracy than the fuzzy *c*-means algorithm.**

***Keywords*— Clustering, fuzzy *c*-means, intrusion detection, KDD Cup 1999 data set, outlier detection**

I. INTRODUCTION

As defined in [1], intrusion detection is the process of monitoring the events occurring in a computer network and analyzing them for signs of intrusions. It is also defined as attempts to compromise the confidentiality, integrity, availability, or to bypass the security mechanisms of a computer network. Anomaly intrusion detection systems (IDSs) aim at distinguishing an abnormal activity from an ordinary one.

The current state of computer networks is vulnerable; they are prone to an increasing number of attacks. These attacks are seldom previously seen. It is very hard to detect them before subsequent damage is done. Therefore, securing such a network from unwanted malicious traffic is of prime concern.

In this paper, a two-phase method for intrusion detection, called 2PID, is proposed. The Knowledge Discovery in Databases (KDD) Cup 1999 data set [2], which has been utilized extensively for development of IDSs, is used as a representative sample of data.

The rest of the paper is organized as follows: Section II introduces the proposed method. Section III describes the experimental setup. Section IV provides the results, and Section V concludes the paper.

II. METHOD DESCRIPTION

In this section, we first review fuzzy *c*-means (FCM) clustering and distance-based outlier detection. Then, we present our proposed method.

A. FCM Clustering

Clustering is an unsupervised classification mechanism where a set of patterns (data), usually multidimensional, are classified into groups (clusters) such that members of one group are similar according to a predefined criterion [3].

FCM is an unsupervised fuzzy clustering algorithm that has been applied successfully to a number of problems involving feature analysis, clustering, and classifier design. It takes unlabeled intrusion data points and tries to group them according to their similarity; points assigned to the same cluster have high similarity, while the similarity between points assigned to different clusters is low [4].

The FCM algorithm partition a set of N patterns $\{X_k\}$ into c clusters by minimizing the objective function

$$J = \sum_{k=1}^{N}\sum_{i=1}^{c}(\mu_{ik})^{m'}\|X_k - m_i\|^2 \quad (1)$$

where $1 \le m' < \infty$ is the fuzzifier, m_i is the i^{th} cluster center, $\mu_{ik} \in [0,1]$ is the membership of the k^{th} pattern to it, and $\|\cdot\|$ is the distance norm. The parameters m_i and μ_{ik} are calculated as

$$m_i = \frac{\sum_{k=1}^{N}(\mu_{ik})^{m'} X_k}{\sum_{k=1}^{N}(\mu_{ik})^{m'}}, \quad (2)$$

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m'-1}}} \tag{3}$$

with $d_{ik} = \|X_k - m_i\|^2$, subject to $\sum_{i=1}^{c} \mu_{ik} = 1$ and $0 < \sum_{k=1}^{N} \mu_{ik} < N$. The algorithm proceeds as follows [5]:

(i) Pick the initial means m_i, $i = 1, 2, ..., c$. Choose the values for the fuzzifier m' and the threshold ε. Set the iteration counter $t = 1$;

(ii) Compute μ_{ik} for *c* clusters and *N* data points, by (3);

(iii) Update m_i by (2);

(iv) Repeat steps (ii) and (iii), by incrementing *t*, until $\|\mu_{ik}(t) - \mu_{ik}(t-1)\| > \varepsilon$.

B. Distance-Based Outlier Detection

Outlier is defined as an observation that appears to be inconsistent with other observations in a data set. Many data-mining algorithms try to minimize the influence of outliers on the final model, or to eliminate them in the preprocessing phases. Outlier detection and potential removal from the data set can be described as a process of the selection of *L* out of *N* samples that are considerably dissimilar, exceptional, or inconsistent with respect to the remaining data.

Distance-based technique is a class of outlier-detection method. The basic computational complexity of this technique is the evaluation of distance measures between all samples in a given data set. Then, a sample in a data set $\{X_k\}$ is an outlier if at least a fraction *p* of the samples in $\{X_k\}$ lies at a distance greater than *r*. Clearly, the criterion for outlier detection is based on *p* and *r*. These two parameters may be given beforehand using knowledge about the data. Further details are in [6].

C. Proposed Method

The 2PID consists of two phases. In the first phase, a set of patterns are classified by FCM clustering. In the second phase, outliers are constructed by a distance-based technique, and a class label is assigned to each pattern.

Binary classification is the task of classifying the members of a given data set into two groups on the basis of whether they have some property or not. The binary classification task in the context of intrusion detection is to differentiate between normal connections and attack situations. In this paper, we focus on such a task.

III. EXPERIMENTAL SETUP

A. KDD Cup 1999 Data Set

The data set provided for the 1999 KDD Cup was originally prepared by MIT Lincoln labs for the 1998 Defense Advanced Research Projects Agency (DARPA) Intrusion Detection Evaluation Program, with the objective of evaluating research in intrusion detection, and it has become a benchmark data set for the evaluation of IDSs. Attacks fall into four main categories:

- Denial of service (DoS), where an attacker makes some computing or memory resource too busy or too full to handle legitimate requests, thus denying legitimate users access to a machine, e.g., SYN flood;
- Remote to local (R2L), where an attacker sends packets to a machine over a network, then exploits machine's vulnerability to illegally gain local access as a user, e.g., guessing password;
- User to root (U2R), where 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, e.g., buffer overflows;
- Probing, where an attacker scans a network to gather information or find known vulnerabilities, e.g., port scanning.

The KDD Cup 1999 data set has a huge number of duplicated records as shown in Table I on the next page. This data set lies with the distribution of its five classes. The DoS attack comprises 79.24% in training and 73.90% in testing, respectively. Meanwhile, normal connection consists of 19.69% in training and 19.48% in testing, respectively. This imbalance makes it very difficult to train classifiers on the training set, and results in having extremely poor detection rates. In this paper, we use a subset of the original data set which consists of distinct records only.

Table I: **Data Distribution and Ratio in the Original Data Set**

Class	Training		Testing	
	Amount of Data	**Ratio (%)**	**Amount of Data**	**Ratio (%)**
Normal	97,278	19.69	60,593	19.48
DoS	391,458	79.24	229,853	73.90
R2L	1,126	0.23	16,189	5.20
U2R	52	0.01	228	0.07
Probing	4,107	0.83	4,166	1.34
Total	494,021	100	311,029	100

B. Data Preprocessing

Data preprocessing has to be undertaken before we could do any experiment. It is carried out in two steps. The first step involves mapping symbolic-valued attributes to numeric-valued attributes. The second step implements non-zero numerical features.

The redundancy in the KDD Cup 1999 data set is surprisingly high. By deleting the repeated data, the size of the data set is reduced from 311,029 to 77,291 as shown in Table II.

Table II: **Data Distribution and Ratio in the Reduced Data Set**

Class	**Amount of Data**	**Ratio (%)**
Normal	47,913	61.99
DoS	23,568	30.49
R2L	2,913	9.77
U2R	215	0.27
Probing	2,682	3.47
Total	77,291	100

IV. RESULTS

Standard measures which were developed for evaluating IDSs include detection rate (DTR), false positive rate (FPR), and overall accuracy (OA). These three performance metrics may be defined as follows [7]:

$$\text{DTR} = \frac{\text{TP}}{\text{TP+FN}} \times 100\%, \tag{4}$$

$$\text{FPR} = \frac{\text{FP}}{\text{TN+FP}} \times 100\%, \tag{5}$$

$$\text{OA} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \times 100\%, \tag{6}$$

where TP, TN, FP, and FN are the numbers of malicious executables correctly classified as malicious, benign programs correctly classified as benign, benign programs falsely classified as malicious, and malicious executables falsely classified as benign, respectively. An IDS requires high DTR, low FPR, and high OA.

The block diagram of our experiment is shown in Fig.1 on the next page. We consider all attacks as a whole, and all 41 features are shown in Table III on the next page. We choose $c = 2,\ p = 230,$ and $r = 1.5.$ The DTRs, FPRs, and OAs for the FCM and the 2PID are shown in Table IV. Obviously, the 2PID yields a higher DTR and a greater OA than the FCM, while the FPRs for both methods are equal. This demonstrates the effectiveness of our proposed method.

Table IV: **Result of the Experiment**

Method	**Detection Rate (DTR)**	**False Positive Rate (FPR)**	**Overall Accuracy (OA)**
FCM	81.07%	2.50%	91.26%
2PID	90.35%	2.50%	94.78%

V. CONCLUSION

This paper has proposed a two-phase approach to intrusion detection, where the KDD Cup 1999 data set has been considered. The experimental results have shown that the proposed method is superior to the FCM. In future work, we plan to include a feature selection algorithm to help build efficient and practical intrusion detection.

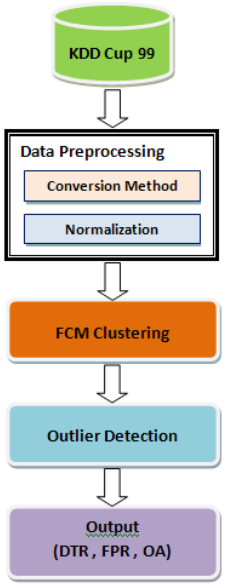

Fig. 1 Block diagram of the experiment

Table III: **Feature Description of the KDD Cup 1999 Data Set**

Feature Name	**Description**	**Type**
1. Duration	Length (number of seconds) of the connection	Continuous
2. Protocol type	Type of the protocol, e.g. tcp, udp, etc.	Discrete
3. Service	Network service on the destination, e.g., http, telnet, etc.	Discrete
4. Flag	Normal or error status of the connection	Discrete
5. Src_bytes	Number of data bytes from source to destination	Continuous
6. Dst_bytes	Number of data bytes from destination to source	Continuous
7. Land	1 if connection is from/to the same host/port; 0 otherwise	Discrete
8. Wrong_fragment	Number of "wrong" fragments	Continuous
9. Urgent	Number of urgent packets	Continuous
10. Hot	Number of "hot" indicators	Continuous
11. Num_failed_logins	Number of failed login attempts	Continuous
12. Logged_in	1 if successfully logged in; 0 otherwise	Discrete
13. Num_compromised	Number of "compromised" conditions	Continuous
14. Root_shell	1 if root shell is obtained; 0 otherwise	Continuous
15. Su_attempted	1 if "su_root" command attempted; 0 otherwise	Continuous
16. Num_root	Number of "root" accesses	Continuous
17. Num_file_creations	Number of file creation operations	Continuous
18. Num_shells	Number of shell prompts	Continuous
19. Num_access_files	Number of operations on access control files	Continuous
20. Num_otbound_cmds	Number of outbound commands in an ftp session	Continuous
21. Is_host_login	1 if the login belongs to the "hot" list; 0 otherwise	Discrete
22. Is_guest_login	1 if the login is a "guest" login; 0 otherwise	Discrete
23. Count	Number of connections to the same host as the current connection in the past two seconds	Continuous
24. Srv_count	Number of connections to the same service as the current connection in the past two seconds	Continuous
25. Serror_rate	% of connections that have "SYN" errors	Continuous
26. Srv_serror_rate	% of connections that have "SYN" errors	Continuous
27. Rerror_rate	% of connections that have "REJ" errors	Continuous

Table III (Continued): **Feature Description of the KDD Cup 1999 Data Set**

Feature Name	Description	Type
28. Srv_rerror_rate	% of connections that have "REJ" errors	Continuous
29. Same_srv_rate	% of connections to the same service	Continuous
30. Diff_srv_rate	% of connections to different services	Continuous
31. Srv_diff_host_rate	% of connections to different hosts	Continuous
32. Dst_host_count	Count for destination host	Continuous
33. Dst_host_srv_count	Srv_count for destination host	Continuous
34. Dst_host_same_srv_rate	Same_srv_rate for destination host	Continuous
35. Dst_host_diff_srv_rate	Dif_srv_rate for destination host	Continuous
36. Dst_host_same_srv_port_rate	Same_src_port_rate for destination host	Continuous
37. Dst_host_srv_diff_host_rate	Diff_host_rate for destination host	Continuous
38. Dst_host_serror_rate	Serror_rate for destination host	Continuous
39. Dst_host_srv_serror_rate	Srv_serror_rate for destination host	Continuous
40. Dst_host_rerror_rate	Rerror_rate for destination host	Continuous
41. Dst_host_srv_rerror_rate	Srv_serror_rate for destination host	Continuous

REFERENCES

[1] R. Bace and P. Mell, "Intrusion Detection Systems," NIST Special Publications on Intrusion Detection Systems. SP 800.31, Nov. 2001.

[2] KDD Data Set. (1999) [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[3] M. K. Pakhira, "A modified *k*-means algorithm to avoid empty clusters," *International Journal of Recent Trends in Engineering*, vol. 1, no. 1, pp. 220-226, May 2009.

[4] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 222-232, Feb. 1987.

[5] R. Jensen and Q. Shen, "Fuzzy-rough data reduction with ant colony optimization," *Fuzzy Sets and Systems*, vol. 149, pp. 5-20, 2005.

[6] M. Kantardzic, *Data Mining: Concepts, Models, and Algorithms*, New Jersey: IEEE Press, 2003.

[7] W. Chimphlee, M. N. M. Sap, A. H. Abdullah, S. Chimphlee, and S. Srinoy, "Anomaly detection of intrusion based on integration of rough sets and fuzzy *c*-means," *Journal of Information Technology*, vol. 17, no. 2, pp. 1-14, Dec. 2005.