Anomaly-Based Intrusion Detection using Fuzzy Rough Clustering

Witcha Chimphlee[1], **Abdul Hanan Abdullah**[2], **Mohd Noor Md Sap**[2],
Surat Srinoy[1], and **Siriporn Chimphlee**[1]

[1]Faculty of Science and Technology, Suan Dusit Rajabhat University
E-mail : {witcha_chi,surat_sri,siriporn_chi}@dusit.ac.th
[2]Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia
E-mail : {hanan, mohdnoor}@fsksm.utm.my

Abstract

It is an important issue for the security of network to detect new intrusion attack and also to increase the detection rates and reduce false positive rates in Intrusion Detection System (IDS). Anomaly intrusion detection focuses on modeling normal behaviors and identifying significant deviations, which could be novel attacks. The normal and the suspicious behavior in computer networks are hard to predict as the boundaries between them cannot be well defined. We apply the idea of the Fuzzy Rough C-means (FRCM) to clustering analysis. FRCM integrates the advantage of fuzzy set theory and rough set theory that the improved algorithm to network intrusion detection. The experimental results on dataset KDDCup99 show that our method outperforms the existing unsupervised intrusion detection methods

1. Introduction

The information protections are often used to protect computer system as the first step of defence. Intrusion detection is the second step for secure defence behind firewall, which can monitor network in the precondition of not affecting network performance. Intrusion detection is the whole process that audits, tracks, identifies and detects the unauthorized accesses or abnormal phenomena, actions and events in the system [1].

The ideal Network Intrusion Detection System will efficiently and effectively classify network traffic between benign and belligerent. A great deal of research and work in Network Intrusion Detection involves the development of attack signatures.

An intrusion detection system (IDSs) is an effective tool for determining whether unauthorized users are attempting to access, have already accessed, or have compromised the network. It is important to find out intrusion quickly and effectively. IDSs may be some software or hardware systems that monitor the different events occurring in the actual network and analyze them for signs of security threats.

Most of machine learning approaches are based on *supervised* learning, and have following problems [2]: 1) a large volume of training data should be collected and classified manually; 2) the performance of the IDS depends on the quality of the training data; 3) a training phase with the huge data is computationally expensive and can not be performed in an incremental manner; 4) it is difficult to detect new intrusions which are not trained. Recently, the *clustering* algorithms based on *unsupervised* learning have been proposed for IDS to overcome these problems [2-5]. Unsupervised learning is very beneficial for intrusion detection domain, since the labeled data is expensive while unlabeled data can be obtained very easily from log files and audit files [6].

A host-base IDS adds a targeted layer to security to particularly vulnerable or essential systems, it is installed on an individual system and monitors audit trails and system logs for suspicious behaviors; a network-based IDS monitors the LAN network traffic, packet by packet, in real time to determine whether traffic conforms to predetermined attack signatures [7].

The basic premise for anomaly detection is that there is intrinsic and observable characteristic of normal behavior that is distinct from that of abnormal behavior. Three main parts in anomaly detection system are: feature selection, model of normal behavior, and comparison [7].

The rest of the paper is organized as follows: Section 2 briefly reviews related work. Section 3 describes the Fuzzy Rough Clustering. Section 4

2006 International Conference on Hybrid Information Technology (ICHIT'06)
0-7695-2674-8/06 $20.00 © 2006 IEEE

presents the Anomaly Detection Approach Based on Fuzzy Rough Clustering. Section 5 describes the details of our experiments, and analysis of results. Section 6 makes some conclusions and outlines some issues for future work.

2. Related work

Since almost all activities are logged on a system, it is possible that a manual inspection of these logs would allow intrusions to be detected. It is important to analyze the audit data even after an attack has occurred, for determining the extent of damage occurred, this analysis helps in attack track back and also helps in recording the attack patterns for future prevention of such attacks [8].

There are lots of researches on the fuzzy clustering. However, most of them focus on the optimization on some fuzzy clustering algorithms or application in some special cases. In more recent work, clustering, majority of work done is intended to optimize clustering. Therefore, we present a feasible detection method based on fuzzy rough clustering and high accuracy.

Without labeled information, it is difficult to distinguish the true outliers from the low density normal points. Many existing anomaly detection algorithms consider all the low density normal points to be outliers, while others would consider the outliers to be normal [9]. Most of the anomaly detection algorithms require the training datasets to be free of attacks. However, clean data entails considerable difficulty for removal of all attacks, including new attacks [10].

There are two major approaches in intrusion detection: anomaly detection and misuse detection.

2.1 Misuse detection

The idea of misuse detection is to represent attacks in the form of a pattern or a signature so that the same attack can be detected and prevented in the future. These systems can detect many or all known attack patterns, but they are of little use for detecting naïve attack methods [8]. Pattern-matching solutions primarily use misuse detection. They employ a library of signatures of misuse, which are used to match against network traffic. The weaknesses of these systems are: variants, false positives, false negatives, and data overload. Since they rely on signatures, a new variant of an attack can be created to evade detection. Additionally, the signatures themselves can create false positives if they are not written correctly, or if the nature of the attack is difficult to isolate from normal traffic characteristics [11]. Earlier studies have utilized a rule-based approach for intrusion detection, but had a difficulty in identifying new attack or attacks that had no previously describe patterns [12].

2.2 Anomaly detection

The idea of anomaly detection is to build a normal activity profile for a system. Anomalous activities that are not intrusive are flagged as intrusive, though they are false positives. Actual intrusive activities that go undetected are called false negatives. This is a serious issue, and is far more serious than the problem of false positives [8]. Anomalies or outliers are aberrant observations whose characteristics deviate significantly from the majority of the data or any events that significantly deviate from normal activity are considered to be suspicious.

The main advantage with anomaly intrusion algorithms is that they can detect new forms of attacks, because these new intrusions will probably deviate from the normal behavior [13, 14]. Most of the commercial and freeware IDS tools are signature based. Such tools can only detect known attacks previously described by their corresponding signatures. The signature database should be maintained and updated periodically and manually for new attacks. For this reason, many data mining and machine learning algorithms are developed to discover new attacks that are not described in the training labeled data [14].

Many intrusion detection approaches have been proposed which include statistical [13], machine learning [15], data mining [16] and immunological inspired techniques [9].

3. Fuzzy Rough Clustering

C-means (HCM) assigns a label to an object definitely; the membership value is 0 or 1. While fuzzy C-means (FCM) maps a membership over the arrange 0 to 1; each object belongs to some or all of the clusters to some fuzzy degrees. Rough c-means (RCM) classify the object space into three parts, lower approximation, boundary and negative region. Then different weighting values are taken in computing the new centers, respectively.

All the objects with RCM in lower approximation take the same weight and all the objects in boundary take another weighting index uniformly. In fact, the objects in boundary regions have different influence on the centers and clusters. So different weighting should be imposed on the objects. The fuzziness membership should be imposed on the objects in boundary.

Define that membership function is given by [17].

2006 International Conference on Hybrid Information Technology (ICHIT'06)
0-7695-2674-8/06 $20.00 © 2006 IEEE

$$u_{ik} = \begin{cases} 1, & x_k \in \underline{A}(v_i) \\ \dfrac{1}{\sum_{j=1}^{c} \left(\dfrac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}}}, & x_k \in \overline{A}(v_i) \quad i = 1,2,...,c; k = 1,2,...,N \end{cases} \quad \textbf{(1)}$$

The new centers are calculated by

$$v_i = \frac{\sum_{k=1}^{N} (u_{ik})^{m_{x_k}}}{\sum_{k=1}^{N} (u_{ik})^m}, i = 1,2,...,c. \quad \textbf{(2)}$$

The objective function used is

$$J_m(u,v) = \sum_{k=1}^{N} \sum_{i=1}^{c} (u_{ik})^m \| x_k - v_i \|^2 \quad \textbf{(3)}$$

The lower and upper approximations are defined respectively

$$\begin{cases} \underline{B}X = \bigcup \{[x_i]_B \mid [x_i]_B \subseteq X\} \\ \overline{B}X = \bigcup \{[x_i]_B \mid [x_i]_B \bigcap X \neq \phi\} \end{cases} \quad \textbf{(4)}$$

For each object *x* and center point *v*, *D(x,v)* is the distance from *x* to *v*. The differences between $D(x,v_i)$ and $D(x,v_j)$ are used to determine the label of *x*. Let

$D(x, v_j) = \min_{1 \le i \le c} D(x, v_i)$ and

$T = \{\forall i, i \neq j: \ |D(x,v_i) - D(x,v_j)| \le Threshold\}$.

1. if $T \neq \phi \Rightarrow x \in \overline{A}(v_i), x \in \overline{A}(v_j)$ and $x \notin \underline{A}(v_l), l = 1,2,...,c$
2. If $T = \phi, x \in \underline{A}(v_j)$, and $x \in \overline{A}(v_j)$.

The fuzzy rough c-means (FRCM) can be formulated as follows in (fig**Error! Reference source not found.**). FRCM will partitions the data into two classes: lower approximation and boundary. Only the objects in boundary are fuzzified.

Input: Unlabeled data; number of cluster c, threshold T, exponent index m, stop criterion ε.
Output: membership matrix U.

Step 1. Let l=0, $J_m^{(0)}(u,v) = 0$; randomly make a membership $U_{c\times N}^{l}$;

Step 2. Compute the c centers $v_i^{(l)}, (i = 1,2,...,c)$ with $U_{c\times N}^{l}$ and data set;

Step 3. Compute the $\underline{A}(v_i^{(l)}), \overline{A}(v_i^{(l)}), u_{ij}^{(l+1)} \ v_i^{(l+1)}, (i = 1,2,...,c)$ with $v_i^l, (i = 1,2,...,c)$, and Threshold T,

Step 4. Compute $J_m^{(l+1)}(u,v)$;

Step 5. $\| J_m^{(l+1)}(u,v) - J_m^{(l)}(u,v) \| < \varepsilon$, stop, otherwise, l=l+1, go to step 2.

Figure 1. Fuzzy Rough c-means algorithm

4. Anomaly Detection Approach Based on Fuzzy Rough Clustering

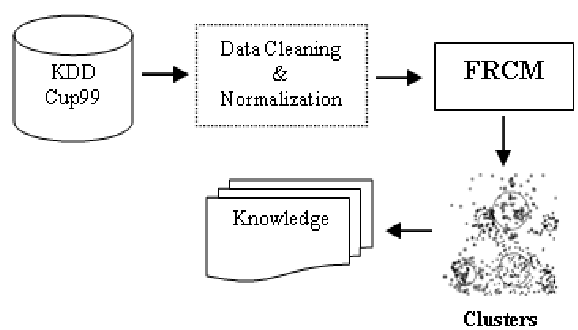

Figure 2. Fuzzy Rough Clustering framework

We take preprocessing in the following three steps:

In the first step, we map symbolic-valued attributes to numeric-valued attributes. Symbolic features like *protocol_type* (3 different symbols-*tcp, udp, icmp*), service (66 different symbols), and flag (11 different symbols) were mapped to integer values ranging from 1 to N where N is the number of symbols. Attack names (like ipsweep, teardrop, etc.) were first mapped to one of the five classes, 0 for Normal, 1 for Probe, 2 for DoS, 3 for U2R, and 4 for R2L.

In the second step, we linearly scale each of these features to the range [0.0, 1.0]. Features having smaller integer value ranges like *duration* [0, 58329], *num_compromised* [0,884], *count* [0,511], *dst_host_count* [0,255] were scaled linearly to the range [0.0, 1.0]. Two features spanned over a very large integer range, namely *src_bytes* [0, 693375640] and *dst_bytes* [0, 5203179] were scaled by logarithmic scaling (with base e) to the range [0.0, 20.4] and [0, 15.5]. For Boolean features having values (0 or 1), they were left unchanged.

The third step involves separating testing dataset into 5 groups.

5. Experiments and Analysis

In our experiments, we perform to classify each of the five classes (normal, probe, denial of service (DoS), user to super-user, and remote to local) of patterns in the KDDCup'99 data. It is shown that using fuzzy rough c-means for clustering. The (training and testing) data set contains 1,011 randomly generated points from the five classes. The distribution of attacks in the KDD Cup dataset is extremely unbalanced. Some attacks are represented with only a few examples, e.g. the *phf* and *ftp_write* attacks, whereas the *smurf* and *neptune* attacks cover millions of records. In general, the distribution of attacks is dominated by probes and *denial-of-service (DoS)* attacks; the most interesting and dangerous attacks, such as compromises, are grossly under represented[18].

5.1 Description of Data Sets

The KDD Cup 1999 [19] data sets are the authoritative testing data sets in current intrusion detection field. This is dataset of features from network packets classified into non-attack and four attack categories. The data are labeled as attack or normal, and furthermore are labeled with an attack type that can be grouped into four broad categories of attacks. The main task of the KDD 99 classifier learning contest was to provide a predictive model able to distinguish between legitimate (normal) and illegitimate (called intrusion or attacks) connections in a computer network. Attacks in the data sets are divided into four main categories:

- DOS (Denial of Service), such as ping of death attack;
- U2R (User to Root), such as eject attack;
- R2U (Remote to User), such as guest attack;
- PROBING, such as port scanning attack.

In the data set, for each TCP/IP connection, there are 41 attributes and a field indicated the intrusion type. Some of the attributes are numerical, such as duration, num_failed_logins, etc. The 10% data set that we used contains 22 different types of intrusions. The total number of this data set is 494,021. In the experiment, we cut the fields that only contain the zero values, the rest have 33 features. We select a testing data set which contained 1,011 records as shown in Table 1

Table 1. Data set for this work

Categories	Number and names of Attack	Total
Normal		500
Probe	Ipsweep(40),nmap(44), Satan(46)	130
DoS	Neptune(396), smurf(404)	180
U2R	buffer_overflow(22), loadmodule (2), perl(2),ps(16),rootkit(13), xterm(13)	68
R2L	ftp_write(3), guess_password(30), imap(1),multihop(18), named(17) ,phf(2), sendmail(17),warezmaste r(32), xlock(9),xsnoop(4)	133
		1,011

5.2 Experiments on Anomaly Detection

The performance measures are calculated from *TP, TN, FP,* and *FN,* which respectively denote the numbers of *true positives* (system traces predicted to be intrusions that are in fact intrusions), *true negatives, false positives*, and *false negatives*. Standard metrics that were developed for evaluating network intrusions usually correspond to detection rate as well as false alarm rate.

1. True Positives (TP), the number of malicious executables correctly classified as malicious;
2. True Negatives (TN), the number of benign programs correctly classified as benign;
3. False Positives (FP), the number of benign programs falsely classified as malicious,
4. False Negative (FN), the number of malicious executables falsely classified as benign.

The measures are based on the formulate

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \textbf{(5)}$$

$$Detection\ rate = \frac{TP}{TP + FP} \quad \textbf{(6)}$$

$$False\ alarm = \frac{FP}{FP + TN} \quad (7)$$

$$Correlation = \frac{(TP.TN) - (FP.FN)}{\sqrt{(TP + FN).(TP + FP).(TN + FP).(TN + FN)}} \quad (8)$$

where the correlation coefficient is a measure of how predictions correlate with actual data. This ranges from -1 to 1 where a correlation coefficient of 1 corresponds to predictions that perfectly match class labels, and a coefficient of 0 corresponds to random guessing [6]. We test the Kmeans and FRCM. Table 3 shows the comparison between Kmeans and FRCM methods

Table 2. Result from experimental

Attack types	# of instance	# record of detection	
		Kmeans	FRCM
normal	500	428	429
probe	130	101	116
DoS	180	96	96
U2R	68	53	59
R2L	133	93	96
Summary	1,011		

Table 3. Comparison Kmeans and FRCM

Detector	Accuracy	Detection rate	False Alarm	Correlation
Kmeans	76.02%	91.81%	16.9%	0.515
FRCM	82.46%	91.45%	24.8%	0.556

The primarily results show that the performance of a proposed approach based on fuzzy rough *c*-means is good. The advantage of using fuzzy logic is that it allows one to represent concepts that could be considered to be in more than one category (or from another point of view – it allows representation of overlapping categories).

6. Conclusion and Future Work

The fuzzy rough clustering algorithm has many advantages. In this paper, we presented a method for clustering to detect normal and abnormal behaviors. We are combining two soft computing methods, a fuzzy rough c-means clustering algorithm (FRCM) that characterizes each class with a positive region, a fuzzy boundary region and a negative region.

It is difficult to make a judgment between normal and abnormal behaviors in certain conditions. So FRCM is introduced to score and distinguish the intrusion behaviors. The result shows that it achieved a good performance that compare with Kmeans methods.

Intrusion detection model is a composition model that needs various theories and techniques. One or two models can hardly offer satisfying results. We plan to apply other theories and techniques to operate in a high accurate and low false alarm rate in intrusion detection in our future work.

References

[1] L. Wang, G. Yu, G. Wang, and D. Wang, "Method of Evolutionary Neural Network-based Intrusion Detection," *Journal NorthEastern University Natural Science*, vol. 23, pp. 107-110, 2002.

[2] H. Lee, Y. Chung, and D. Park, "An Adaptive Intrusion Detection Algorithm Based on Clustering and Kernel-Method," presented at Advances in Knowledge Discovery and Data Mining, 10th Pacific-Asia Conference, PAKDD 2006, Singapore, 2006.

[3] L. Portnoy, E. Eskin, and S. J. Stofo, "Intrusion detection with unlabeled data using clustering," presented at Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001), Philadelphia, 2001.

[4] N. Ye and X. Li, "A Scalable Clustering Technique for Intrusion Signature Recognition," presented at Proceedings of the IEEE Man, Systems and Cybernetics Information Assurance Workshop United States Military Academy West Point, New York, 2001.

[5] Y. Liu, K. Chen, X. Liao, and W. Zhang, "A genetic clustering method for intrusion detection," *Pattern Recognition*, vol. 37, pp. 927-942, 2004.

[6] Y. Wang, J. Wong, and A. Miner, "Anomaly intrusion detection using one class SVM," presented at 5th Annual IEEE Information Assurance Workshop, West Point, New York, 2004.

[7] Y. Liu, D. Tian, and B. Li, "A Wireless Intrusion Detection Method Based on Dynamic Growing Neural Network," presented at Proceedings of the First International Multi-Symposiums on Computer and Computational Sciences (IMSCCS'06), 2006.

[8] S. Mukkamala, A. H. Sung, and A. Abraham, "Intrusion detection using an ensemble of intelligent paradigms," *Journal of Network and Computer Applications*, vol. 28, pp. 167-182, 2005.

[9] J. Gao, H. Cheng, and P.-N. Tan, "A Novel Framework for Incorporating Labeled Examples into Anomaly Detection," presented at 2006 Siam Conference on Data Mining, Bethesda, Maryland, USA 2006.

[10] M. H. Arshad and P. K. Chan, "Identifying Outliers via Clustering for Anomaly Detection," Florida Institute of Technology, Department of Computer Sciences Technical Report CS-2003-19, 2003.
[11] K. Liston, "Intrusion Detection FAQ: Can you explain traffic analysis and anomaly detection?."
[12] K. Ilgun, "USTAT: A Real-Time Intrusion Detection System for UNIX," presented at Proceedings of the 1993 IEEE Symposium on Security and Privacy, 1993.
[13] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. 13, 1987.
[14] Y. Bouzida, F. Cuppens, N. Cuppens-Boulahia, and S. Gombault, "Efficient Intrusion Detection Using Principal Component Analysis," presented at 3ème Conférence sur la Sécurité et Architectures Réseaux (SAR), La Londe, France, 2004.
[15] T. D. Lane, "Machine Learning techniques for the Computer Security of Anomaly Detection," vol. Ph.D.: Purdue University, 2000.
[16] W. Lee, *A data mining framework for constructing features and models for intrusion detection systems*: Columbia University, 1999.
[17] Q. Hu and D. Yu, "An Improved Clustering Algorithm for Information Granulation," presented at 2nd International Conference, FSKD 2005, Changsha, China, 2005.
[18] P. Laskov, K. Rieck, C. Schäfer, and K.-R. Müller, "Visualization of anomaly detection using prediction sensitivity," presented at Proceeding of Sicherheit, 2005.
[19] "KDD Cup 99 Intrusion Detection Datasets, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html," vol. 2006.