A Method for Locating Digital Evidences with Outlier Detection Using Support Vector Machine

Zaiqiang Liu, Dongdai Lin, and Fengdeng Guo
(Corresponding author: Zaiqiang Liu)

State Key Laboratory of Information Security, Institute of Software
Chinese Academy of Sciences, BP.O.Box 8718, Beijing 100080, China (Email: liuzq@is.iscas.ac.cn)

(Received Apr. 6, 2006; revised and accepted July 31, 2006)

Abstract

One of the biggest challenges facing digital investigators is the sheer volume of data that must be searched in locating the digital evidence. How to efficiently locate the evidence relating to the computer crime while maintaining accuracy is becoming a research focus. In this paper, we introduce a two-tier method to automate the process of locating the digital evidence, which first employ a one-class Support Vector Machine (SVM) outlier detector to filter out insignificant records for forensic investigators and then use a group of one-class SVM classifiers (trained with the expert knowledge or interested samples for an investigator based on a different feature vector) to further analyze the output of the outlier detector to improve the accuracy of investigation. The effectiveness of the proposed method for locating digital evidence is demonstrated using the public datasets: KDD Cup99 (Knowledge Discovery and Data-mining) intrusion detection dataset.

Keywords: Data mining, digital forensics, feature calculation, support vector machine

1 Introduction

A digital investigation is a process where investigators develop and test hypotheses that answer questions about digital events. This process is achieved using the scientific method where an investigator develops a hypothesis based on the existing evidence that he finds and then tests the hypothesis by looking for additional evidence that shows the hypothesis is true or false [4]. During the process of a digital investigation, the step of searching for digital evidence to support or refute the hypothesis is one of the most time consuming tasks. Digital evidence is a kind of digital data that contains reliable information that support or refute a hypothesis about the incident being investigated [5]. The most common technique to search for digital evidence is the "string search" based on keywords provided by the investigators. This method is usually simple and effective, but it requires investigators to assemble a list of words specific to the investigated incident beforehand. Recently, Brain Carrier proposed a target definition method using outlier analysis to automate the process of the searching for digital evidence [5], and the results of the experiment show that the method is feasible, but the false positive rate is still high. More research into the false rates of evidence searches is needed to improve the process of digital forensics with automated techniques. In this paper, we propose a new outlier detection method based on SVM to speed up the searching process for digital evidence while improving the accuracy of locating potential evidence.

The remainder of this paper is organized as follows. Section 2 provides the introduction of generalized SVM algorithm. Section 3 introduces the automated technique of locating digital evidence that is based on SVM. Section 4 describes the process and results of experiments that use our proposed methods to find network session that are potential evidence. Finally Section 5 concludes the paper.

2 Support Vector Machine

The SVM is a maximal margin algorithm that is based mainly on work performed by Vladimir N. Vapnik and coworkers, which was presented first in 1992 [1]. The SVM was primarily constructed to solve binary classification problems, and now has been improved to solve multi-classification problems. It has much better qualities than other data mining techniques: good capacity for generalization; less susceptible to overfitting; efficient in dealing with the problem of local optimum, etc.

The basic principle of SVM is to map feature vectors to a high dimensional space and to search a hyperplane that not only separates the training vectors from different classes, but also maximizes this separation by making the margin as large as possible. This can be illustrated by a binary classification problem (see Figure 1), and described as follows:

1) Assume training dataset $T = \{(X_1, Y_1), \cdots (X_i, Y_i)$

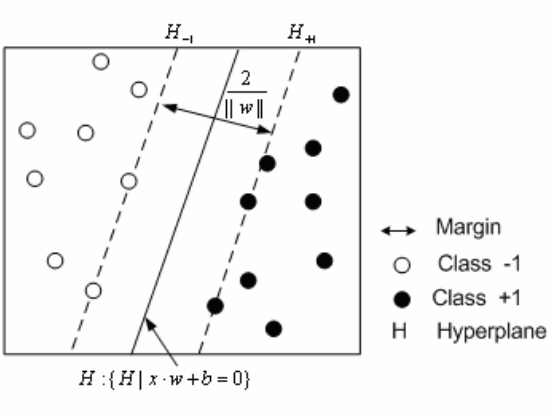


Figure 1: SVM hyperplane

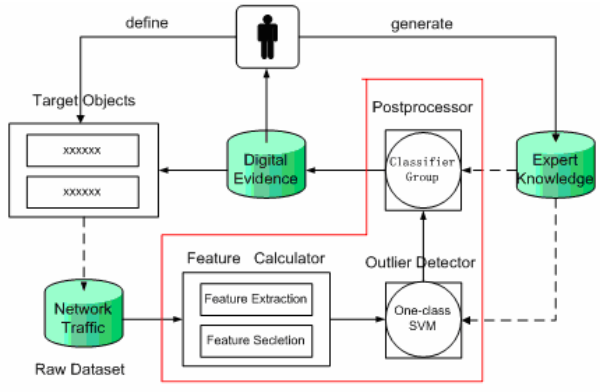


Figure 2: Outlier detection framework for searching for digital evidence

$, \cdots (X_l, Y_l)\} \in (\chi \times \gamma)^l$, where $x \in \chi = R^n, y_i \in \gamma = \{+1, -1\}, i = 1, \cdots, l$;

2) Seek an optimal hyperplane, such as $\{H|x \cdot w + b = 0\}$ in Figure 1, between classes of points such that the distance between the closest points is maximized. It is equivalent to the solution of the following optimization problem:

$$Minimize\ \frac{1}{2}w^T w + C\Sigma_{i=1}^{l}\xi_i$$
$$Subject\ to\ y_i(w^T\Phi(X_i)) \geq 1 - \xi_i, \quad \xi_i > 0.$$

Where training vectors x_i are mapped into a higher dimensional space by the function $\Phi(\bullet)R^n \rightarrow R^m$, $m > n$, that can be linear or nonlinear; $C > 0$ is the penalty parameter of the error term; w is a vector in a high dimensional space R^m;

3) Transfer the above optimization problem into its dual problem. The above optimization problem is equivalent to the dual:

$$Minimize\ \frac{1}{2}\Sigma_{i=1}^{1}\Sigma_{j=1}^{l}y_i y_j \alpha_i \alpha_j K_\Phi(x_i, x_j) + \Sigma_{j=1}^{l}\alpha_j$$
$$Subject\ to\ \Sigma_{i=1}^{l}y_i\alpha_i = 0, \quad 0 \leq \alpha_i \leq C.$$

Where $K_\Phi(x_i, x_j) = (\Phi(x_i) \cdot \Phi(x_i))$ is a kernel function. Calculate and gain the optimal answer $\alpha^* = (\alpha_1^*, \alpha_2^*, \cdots \alpha_l^*)^T$;

4) Calculate w^* and b^*. Here $w^* = \Sigma_{i=1}^{l}y_i\alpha_i^*\Phi(x_i)$, and choose $\{\alpha_j^* \mid 0 \leq \alpha_j^* \leq C\}$ and calculate the $b^* = y_j - \Sigma_{i=1}^{l}y_i\alpha_i^* K_\Phi(x_j, x_i)$;

5) Calculate the decision function:

$$\begin{aligned} f(x) &= sgn((w^* \cdot \Phi(x)) + b^*) \\ &= sgn(\Sigma_{i=1}^{l}y_i\alpha_i^* K_\Phi(x, x_i) + b^*). \end{aligned}$$

The above SVM algorithm is actually a supervised two-class classifier, so if we want to train the SVM, we must have the labelled datasets consisting of both positive (normal) samples and negative (abnormal) samples. It is most impractical in the process of investigation to acquire enough training datasets. Luckily, Scholkopf and his coworkers have proposed a method to adapt the SVM algorithm to the one-class classification problem [12], which is called a one-class SVM algorithm. The basic strategy of a one-class SVM algorithm is to identify "outliers" in normal samples and regard them as anomalous samples, and then the standard classification SVM techniques are employed. Therefore we can use this quality of one-class SVM to build an evidence model with unlabelled datasets that is easier to acquire. Section 3 details the proposed methods about how to use SVM to locate digital evidence.

3 Proposed Methods for Searching Digital Evidence

Usually outlier detection techniques can be applied in conditions as follows: to cleanse the training dataset during the data preprocessing process of data mining; and to gain anomalous data sets of outliers that are often of particular interest to users. The process of searching for digital evidence mainly uses the second aspect of the functions of outlier detection techniques in order to identify potentially interested data or content that guides investigators to further searching. The process often is carried out when digital investigators have collected some known evidence for collecting more relative evidence. But sometimes there is no relative evidence with the current investigated event before investigating, so digital investigators need the guide to locate a piece of digital evidence in the case of being lost during the "looking for a needle" in a bottle of "data". Trying to resolve the above problem, this paper introduces a method of searching digital evidence automatically. Our method employs the one-class SVM technique as an outlier analysis method to find digital evidence from the file system image or captured network traffic dataset, which is relative to the investigated incident in time or space.

Unfortunately, it is difficult for investigators to choose

a point which can keep a low false negative rate while keeping a low false positive rate when using outlier detection techniques. We resolve this problem by using a two-tier approach that combines outlier detection to reduce false negative rate with postprocessor to reduce false positive rate (see Figure 2). Datasets needing investigation are first preprocessed, and attributes describing the target object are extracted from investigated datasets by the Feature Calculator. Then the Outlier Detector is employed to learn the "outlier-ness" of the records of interest to the investigators and to eliminate the records that have a high-probability of being normal. Then the outliers detected by the Outlier Detector and the rest of the records are passed on to Postprocessor which can be regarded as a secondary classifier to further determine whether they are false positives from the outlier detector or potential evidence with the support of existing knowledge consisting of expert knowledge and existing evidence samples. Records outputted by the postprocessor will be labelled as potential evidence samples that can be used as existing knowledge in the next iteration, and passed on to the Target Object unit to define or update the searching targets for further investigation. Repeat the above steps until the evidence of an incident which refutes or supports an investigation hypothesis [3] have been acquired. The formal description of the above components is detailed in the following sections respectively.

3.1 Feature Calculator

Feature Calculator (FC) is responsible for extracting features from the original investigated data source. Feature extraction and selection from the available data is important to the effectiveness of the methods employed because the great capability in selecting the suitable features of a classifier can lead directly to faster training and more accurate results. Usually the selection of what kinds of features depends on the target objects defined or constructed by the forensic investigator. In theory, the more easily to select features, the more specific the target object is. In fact, it is very difficult to define a clear target object before carrying out a forensic investigation of an incident. But it is comparatively easy to identify the category the investigated incident belongs to. So we categorize the target object into five different classes R2L, DOS, Probe, U2R, Normal for network forensics. Where R2L denotes unauthorized access from a remote machine, such as guessing a password; DOS denotes denial-of-service, such as smurf attack; U2R denotes unauthorized access to local superuser privileges, such as various "buffer overflow" attacks; "probe" denotes surveillance and other probing, such as host or port scanning [14].

Under the network environment, there are many traffic features that can be used for intrusion detection or event analysis, such as, source address and port number, destination address and port number, timestamp, etc. Stolfo and his team have researched this topic in-depth and calculated 41 different features in all applied to various incidents. For more detail information about feature calculation, please refer to [9, 14]. However using too many features for various incidents will cause the problem of over-fitting of both the Outlier Detector and the Postprocessor, which will increase the cost of calculation and reduce accuracy. The best way is to select a subset of features used for the investigation. We analyzed various combinations of features for their contribution to the Outlier Detector and the Postprocessor accuracy according to the method proposed in [10]. The result of analyzing about the relationship between incident category and its feature subset is described in Table 1.

3.2 Outlier Detector Using One-class SVM

After extracting and selecting the suitable features, the next step is to eliminate records that have a very high probability of being normal through an outlier detector. Although almost any anomaly detection methods could be applied, we employ a commonly used one-class Support Vector Machine (SVM) with a modified Gaussian (RBF) kernel. The standard RBF kernel function is based on the Euclidean Distance function. One weakness of the basic Euclidean distance function is that if one of the input attributes has a relatively large range, then it can overwhelm the other attributes. For example, if there are two attributes (A1 and A2), and A1 can have values from 1 to 10000, and A2 has values only from 1 to 10, then A2's influence on the distance function could be overwhelmed by A1's influence. Besides this, it can not effectively handle applications with both continuous and nominal attributes. So we redefine the RBF kernel function as:

$$K_\Phi(x,y) = e^{\frac{-\|D(x,y)\|}{\delta^2}}.$$

Where $D(x,y)$ is the Heterogeneous Value Difference Function (HVDM)[15] and defined as:

$$D(x,y) = (\Sigma_{i=1}^{m} d_i^2(x_i,y_i))^{\frac{1}{2}}.$$

Where m is the number of attributes, d_i is the distance function for ith attribute and defined as:

$$d_i(x_i,y_i) = \begin{cases} 1, & x_i \text{ or } y_i \text{ unknown} \\ d_{vdm}(x_i,y_i), & x_i \text{ and } y_i \text{ are nominal} \\ d_{diff}(x_i,y_i), & x_i \text{ and } y_i \text{ are numeric} \end{cases}$$

where $d_{vdm}(x_i,y_i) = \Sigma_{j=1}^{k} \mid \frac{N_{i,x,j}}{N_{i,x}} - \frac{N_{i,y,j}}{N_{i,y}} \mid$, $d_{diff}(x_i,y_i) = \frac{|x_i - y_i|}{4\sigma_i}$, and

- σ_i is the standard deviation of the numeric values of ith attribute;
- $N_{i,x}$ is the number of instances in the training set T that have value x for ith attribute;
- $N_{i,x,j}$ is the number of instances in T that have value x for ith attribute and output class j;

Table 1: Relationship between categories and feature subsets

Category	Feature subset vector	Marker
R2L	[3,5,6,32,33]	FSV(r2l)
DOS	[1,5,6,23,24,25,26,32,36,38,39]	FSV(dos)
Probe	[3,5,6,23,24,32,33]	FSV(probe)
U2R	[2,3,5,6,24,32,33]	FSV(u2r)
Normal	[1,2,3,4,5,6,10,17,23,27,28,29,33,36,39]	FSV(norm)

Note: the numbers in the above "Feature Subset Vector" column are corresponding to the serial number of 41 various quantitative and qualitative features extracted in [14].

- K is the number of output classes in the problem domain.

A one-class SVM uses the above kernel function which transforms the unlabelled examples into a high-dimensional feature space, and learns the support region for "normal" data. In order to maximally separate the "normal" data from the origin via a hyperplane boundary, the one-class SVM needs to resolve the following quadratic programming problem:

$$Minimize\ \frac{1}{2}w^T w + \frac{1}{vl}\Sigma_{i=1}^{l}\xi_i - p$$
$$Subject\ to\ y_i(w \cdot \Phi(x_i)) \geq p - \xi_i, \xi_i \geq 0, \quad i = 1, \cdots, l.$$

Where ξ_i is the slack parameter which associated with each data example and denotes the possibility that some of the training examples can be misclassified [13]; the parameter $v \in (0, 1)$ controls the trade off between maximizing the distance from the origin and containing most of the data in the region created by the hyperplane, and is the upper bound on the ratio of outliers in training dataset [13]; the parameter ρ is the offset, and the distance between the margin and the origin is $\rho \parallel w \parallel$. For a more detailed explanation of SVM algorithm, see Section 2.

The Outlier Detector is trained on unlabelled dataset consisting of feature vector FSV (norm) to gain a generalized profile of "normal" activity. By excluding a large portion of the fringe normal records from the original dataset, the outlier detector can identify, with high confidence, that certain subset of records are not anomalous. The rest of the original dataset will be anomalous, but with high false positives, and actually they are the data points needed to pay attention by investigators. Usually the False Positives (FP) of a classifier is in conflict with its True Positives (TP), so we have to improve the FP rate of the classifier at the cost of its TP rate, and vice versa. In theory the ideal balance point will be the nearest point from the top-left point in the Receiver Operating Characteristic (ROC) curve [15]. See Figure 3, the ideal point will be the point T. However, most of the time the TP rate at the point T is not equal to 1, which means that a portion of samples are misclassified as normal, that is to say, some potential evidence will be missed after the processing of the Outlier Detector. In order to reduce the probability of missing potential evidence samples, we can

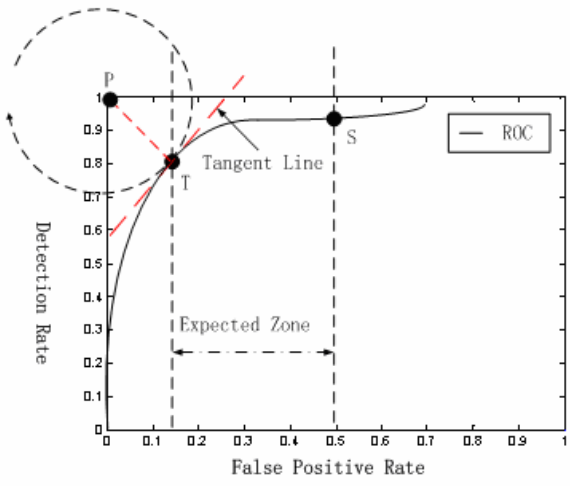


Figure 3: Ideal detection point

select the balance point from the Expected Zone in Figure 3. The Detection Rate of the point S in Figure 3 can be a number equal or near to 100%. Finally the output of the Outlier Detector that is deemed anomalous will be passed on to the Postprocessor for further classification to reduce the false positive rate.

Note that the Outlier Detector needs not create a model for anomalous data, but if there are some known anomalous examples which can be used as training data for the secondary classifier in the Postprocessor, it can improve the accuracy of the outlier detector.

3.3 Postprocessor

The Postprocessor functions as a group of classifiers used to filter false positives from the output of the Outlier Detector. While many artificial intelligence methods could be applied to this task, we still employ one-class SVM algorithm with a modified Gaussian (RBF) kernel to implement a group of classification models trained from anomalous records. For a more detailed explanation of SVM algorithm, see Section 3.2. In order to support investigation on various network incidents, the Postprocessor builds four models with FSV(r2l), FSV(dos), FSV(probe), and FSV(u2r) respectively. An investigator can choose

one of the models (such as FSV(dos) model) to filter the false positives under the support of existing evidence or knowledge (see Figure 4(a)), or the union of multiple models, such as (see Figure 4(b)), to systematically evaluate the output of each classifier under the support of forensic experts.

The output of the Postprocessor will be the potential evidence or evidence snippet, and a forensic investigator can define new target objects or employ other tools (such as keyword searching tool) on the bases of the output for further investigation.

4 Experiments and Results

In this section, we introduce the experiment methods and present initial results of the use of our proposed method to detect outlier in intrusion detection databases. In all experiments we employ a SVM tool called LIBSVM [6] and the modified RBF kernel.

4.1 Dataset

The data for our experiments was prepared by the 1999 DARPA intrusion detection evaluation program from MIT Lincoln Labs [9]. The following experiments are based on the 10% train data subset with 494,021 data records. Each record has 41 attributes for each connection plus one class label, and the class label will only be used for testing, not participation in the classification.

4.2 Methods

Due to the raw dataset being collected in a simulated network attack environment, the proportion of attack instances to normal ones in the KDD training dataset is very high (over 400%). It is almost impossible in a true network environment and it also breaches the basic precondition of efficiently classification of interested instances for the investigators. So we filtered some of attack instances from the original and keep the proportion of attack instances (outliers) in the range of [1%, 1.5%].

The goal of the experiment is to separate the outlier from the above datasets and identify the impact of the Outlier Detector on the secondary classifier both in accuracy and required training data size. In order to achieve the goals, we constructed different training data and test data from each above dataset by means of randomly selecting sub-dataset with certain ratio of normal data to attack data, and then different experiments were carried out for each class which was regarded as an outlier or potential interested instances by forensic investigators. The test datasets constructed for evaluating the performance of our proposed methods is described as follows:

- DS(r2l): DataSet(normal) ∪ DataSet(r2l);
- DS(dos): DataSet(normal) ∪ DataSet(dos);
- DS(probe): DataSet(normal) ∪ DataSet(probe);
- DS(u2r): DataSet(normal) ∪ DataSet(u2r);
- DS(norm): DataSet(normal) ∪ DataSet(r2l) ∪ DataSet(u2r) ∪ DataSet(dos) ∪ DataSet(probe).

Besides this, in order to build the classification models for the Postprocessor, we randomly selected different number (K) of attack samples from the filtered attack samples, and trained the classifiers.

In order to measure the performance of the proposed method, the ROC curve is used. The ROC curve is a plot of detection accuracy against the false positive rate [15]. It can be obtained by varying the detection threshold. Detection rate and false positive rate may be defined as follows:

$$\begin{aligned} \textit{Detection rate} &= TP/(TP+FN) \\ \textit{False positive rate} &= FP/(FP+TN). \end{aligned}$$

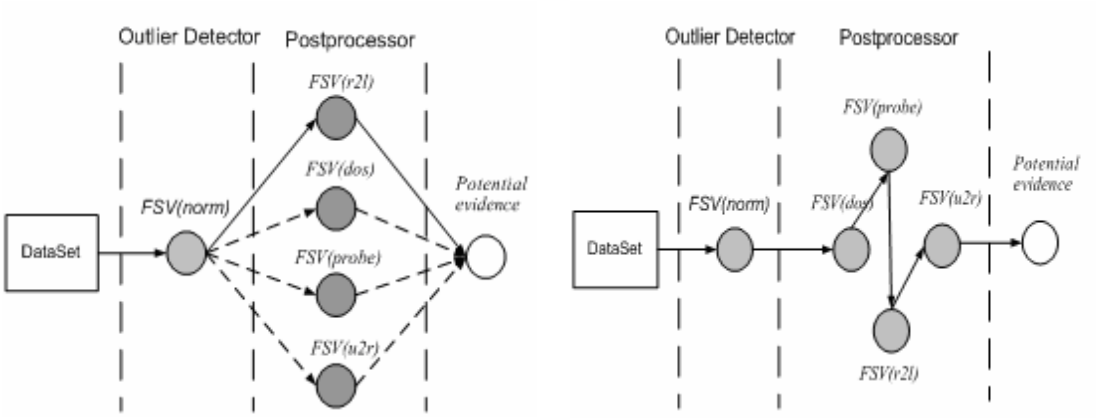
Where TP denotes true positives, FN denotes false negatives, FP is false positives and TN is true negatives. Besides this, both the Outlier Detector and the Postprocessor employ the SVM algorithm, and we hope that our SVMs can detect outliers with high accuracy, while classifying normal examples with great confidence. This is controlled to a great degree by the parameter v. In order to reduce the complexity of the experiments, all experiments were performed with constant value of v (0.125) selected by a great deal of experiments.

4.3 Analysis of Results

Five experiments have been done with the datasets generated in the above section. The results of experiments in Division Mode are presented in Figure 5(a-d).

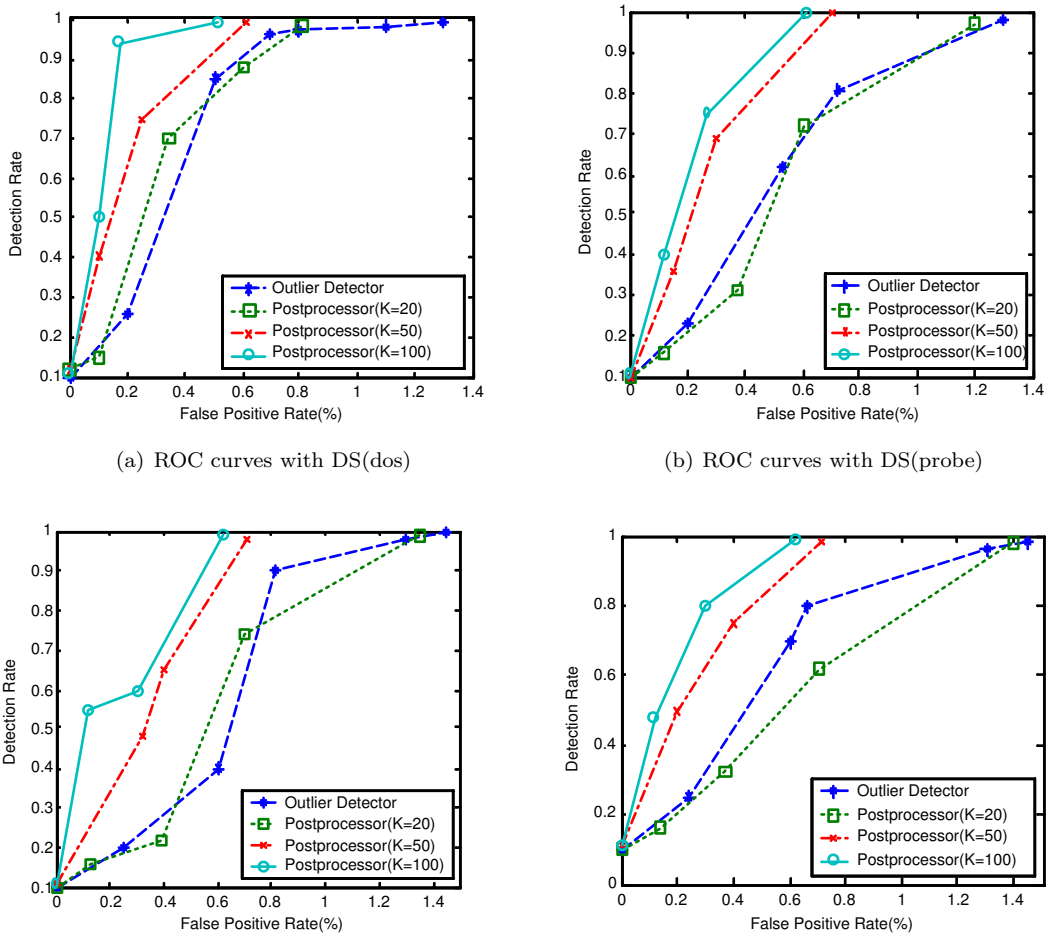
In the above figures, the parameter K is the number of corresponding interested samples for training the Postprocessor classifier. From the figures, we can see that the Outlier Detector has good performance in rate of detection, especially in Figure 5(a) the Detection Rate reaches 92% while FP rate 0.43%. Besides this, when the results of the Detector Outlier were pipelined to the Postprocessor which was trained with the fewer training samples (about $K = 50$), the Postprocessor greatly reduces the false positives generated by the Outlier Detector while keeping the same detection rate. For the experiment in Union Mode, we got a similar conclusion (see Figure 6). From the results of these experiments we can also conclude that the proposed methods can keep good accuracy when automating the analysis process of locating the potential evidence, and the proposed method could be used in the practice of network forensic investigation.

From all experiments, we also noted that when the parameter K is smaller, the false positives would become even worse than the results of the Outlier Detector. In order to further evaluate the impact of the parameter K on the False Positive, we did a group of evaluating experiments with DS(dos), DS(probe), DS(r2l), DS(u2r) and



(a) Division Mode (b) Union Mode

Figure 4: Running mode of the postprocessor



(a) ROC curves with DS(dos) (b) ROC curves with DS(probe)

(c) ROC curves with DS(r2l) (d) ROC curves with DS(u2r)

Figure 5: ROC curves in division mode

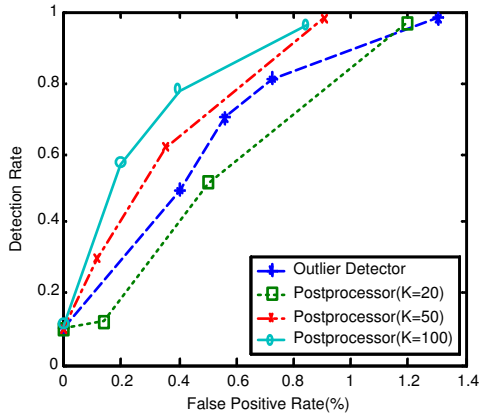


Figure 6: ROC curves in union mode with DS(norm)

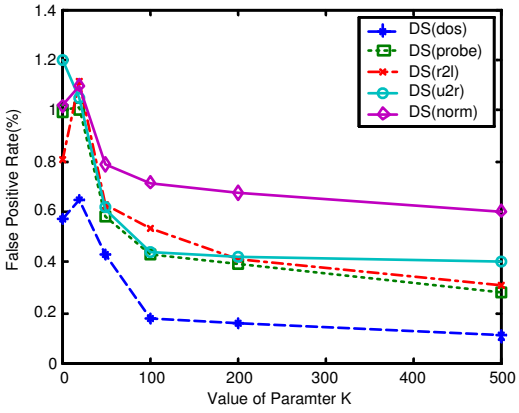


Figure 7: Relation curves between parameter K and false positives

DS(norm) datasets individually. The results of experiments are presented in Figure 7 which shows the relationship between the parameter K and the False Positive rate when the detection rate is a constant (90%). From the Figure 7, we can draw the conclusion that the increase of the parameter K will help decrease the False Positive at the very start, and this trend will tend to smooth with the increase of K. So we can conclude further that the parameter K should have an ideally critical value which will benefit the decrease of the False Positive to a great degree. In our experiments, the critical value of K is about 100. The critical value of K depends on both the interested type of event and the quality of training data. In practice, the selection of K will become even worse, and how to determine the value of K will be a challenging topic in theory.

5 Conclusions

Digital forensics presents a great number of new and interesting challenges to computer security researchers. Especially the huge volume of data to be analyzed often frustrated the forensic investigators. In the paper, we proposed the forensic framework to automate and speed up the process of locating the potential evidence in the network forensics, which based on the one-class SVM algorithm with a modified Gaussian (RBF) kernel to implement a group of classification models to screen out the potential evidence. Besides this, we used a two-tier approach that combines outlier detection to reduce false negative rate with postprocessor to reduce false positive rate. From the results obtained in these early experiments, we believe that the proposed method has promise for the automation of digital forensic investigation.

Further experiments should be conducted with different learning algorithms and paradigms to allow performance comparisons with the proposed approach. Since the Postprocessor still requires an investigator to supply the training data or expert knowledge to build classification model which increases the burden of the investigator, therefore we suggest that an effective unsupervised-learning method could be employed. This is one of our future goals.

References

[1] Be. E. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," *Computational Learing Theory*, pp. 144-152, 1992.

[2] C. Campbell and K. P. Bennett, "A linear programming approach to novelty detection," *Advances in Neural Information Processing Systems*, vol. 14, pp. 395-401, 2001.

[3] B. D. Carrier and E. H. Spfford, "Defining event reconstruction of a digital crime scene," *Journal of Forensic Sciences*, vol. 49, no. 6, pp. 1291-1298, 2004.

[4] B. Carrier, *File System Forensic Analysis*, Addison Wesley Professional, 2005.

[5] B. D. Carrier and E. H. Spafford, "Automated digital evidence target definition using outlier analysis and existing evidence," in *2005 Digital Forensic Research Workshop (DFRWS)*, 2005.

[6] C. C. Chang and C. J. Lin, *LIBSVM: A Library for Support Vector Machines*, 2001. (http://www.csie.ntu.edu.tw/ cjlin/libsvm)

[7] M. Davy, A. Gretton, A. Doucet, and P. J. W. Rayne, "Optimised support vector machines for nonstationary signal classification," *IEEE Signal Processing Letters*, vol. 9, pp. 442-445, 2002.

[8] F. Desobry and M. Davy, "Support vector-based online detection of abrupt changes," in *Proceedings of ICASSP*, vol. 4, pp. 872-875, 2003.

[9] W. Lee and S. J. Stolfo. "Data mining approaches for intrusion detection," in *Proceedings of the 7th USENIX Security Symposium*, pp. 79-93, 1998.

[10] S. Mukkamala and A. H. Sung, "Identifying significant features for network forensic analysis using artificial intelligent techniques," *International Journal of Digital Evidence*, vol. 1, no. 4, pp. 1-17, 2003.

[11] F. Provost and T. Fawcett, "Robust classification for imprecise environments," *Machine Learning*, vol. 42, pp. 203-231, 2001.

[12] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443-1471, 2001.

[13] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond* MIT Press, Cambridge, MA, 2002.

[14] S. J. Stolfo, W. Fan, and W. Lee, "Cost-based modeling and evaluation for data mining with application to fraud and intrusion detection: Results from the JAM projec", in *DARPA Information Survivability Conference*, pp. 130-144, 2000.

[15] D. R. Wilson, and T. R. Martinez, "Improved heterogeneous distance functions," *Journal of Artificial Intelligence Research*, no. 6, pp. 1-34, 1997.

Zaiqiang Liu is now a Ph.D candidate at the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences. His research interests include Digital Forensics and Network Security. E-mail address: zaiqiangliu@gmail.com.

Dongdai Lin is now a full time research professor and deputy director of State Key Labortory of Information Security, Institute of Software of the Chinese Academy of Sciences. He received his B.S. degree in mathematics from Shandong University in 1984, and the M.S. degree and Ph. D degree in coding theory and cryptology at Institute of Systems Science of the Chinese Academy of Sciences in 1987 and 1990 respectively. His current research interests include cryptology, information security, grid computing, mathematics mechanization and symbolic computations.

Fengdeng Guo is now a full time research professor and director of State Key Labortory of Information Security, Institute of Software of the Chinese Academy of Sciences. His current research interests include cryptology, information security.