A Framework for Adaptive Anomaly Detection Based on Support Vector Data Description*

Min Yang, HuanGuo Zhang, JianMing Fu, and Fei Yan

School of Computer, State Key Laboratory of Software Engineering,
Wuhan University, Wuhan 430072, Hubei, China
yangm75@hotmail.com

Abstract. To improve the efficiency and usability of adaptive anomaly detection system, we propose a new framework based on Support Vector Data Description (SVDD) method. This framework includes two main techniques: online change detection and unsupervised anomaly detection. The first one enables automatically obtain model training data by measuring and distinguishing change caused by intensive attacks from normal behavior change and then filtering most intensive attacks. The second retrains model periodically and detects the forthcoming data. Results of experiments with the KDD'99 network data show that these techniques can handle intensive attacks effectively and adapt to the concept drift while still detecting attacks. As a result, false positive rate is reduced from 13.43% to 4.45%.

1 Introduction

Intrusion detection is a necessary complement to traditional intrusion prevention techniques to guarantee network security. There are two general approaches for intrusion detection: misuse detection and anomaly detection [1]. Compared with misuse detection, anomaly detection has the advantage that it can detect new types of attacks. However, at the same time, it suffers from high false alarm especially when normal behavior changes over time. In practice, users, networks or system activities cannot be invariant when environment changes over time. This phenomenon is called concept drift [2]. To guarantee the accuracy of adapting to concept drift while still recognizing anomalous activities, adaptive anomaly detection systems have to retrain and update their models with online or newly collected data frequently [3].

Unsupervised learning algorithms, which train models with unlabelled data, are promising for adaptive anomaly detection and have been studied by researchers in recent years [3–6]. In [3], a general adaptive model generation system to anomaly detection is presented, which uses a probability-based algorithm

* Supported by the National Natural Science Foundations of China under Grant No.90104005 and No.66973034, supported by the National 863 Project under Grant No.2002aa141051, supported by the Doctoral Science Foundation of Ministry of Education under Grant No.20020486046

for building models over noisy data periodically. SmartSifer, an online unsupervised learning algorithm for anomaly detection based on a probabilistic model, adjusts the model after each input datum [4]. More recently, several different unsupervised learning algorithms are applied to anomaly detection, including cluster-based algorithm, k-nearest neighbor based algorithm, LOF approach, and one-class SVM algorithm [7]. The work most similar to our SVDD-based unsupervised anomaly detection is one-class SVM based anomaly detection. Those algorithms as well as previous probabilistic based algorithms [3, 4] make an important assumption of attack ratio that attacks can be taken as outliers because they are rare and qualitatively different from normal data. Therefore, these algorithms can use real time data to constantly update or periodically retrain their models directly. However,the assumption of attack ratio, i.e. normal data greatly outnumber the attacks, limits the application of these algorithms in practice because the number of large-scale DoS attacks and probing attacks has been increasing alarmingly over the past few years. As a result,the assumption does not hold when a burst of intensive attacks causes a large number of anomaly instances in a short time.

In this paper, we present a new framework for adaptive anomaly detection, which extends traditional unsupervised method and overcomes the limitations of the assumption of attack ratio. In the framework, we introduce the SVDD algorithm to anomaly detection. Also, an SVDD-based online change detection algorithm is presented to distinguish changes caused by intensive attacks from concept drift. With the aid of change detection algorithm, intensive attacks is filtered first, and then model retraining is realized safely.

The rest of this paper is organized as follows. In section 2, we describe the SVDD algorithm and introduce the change point detect algorithm; based on these algorithms, we then present the SVDD-based adaptive anomaly detection framework. In section 3, we discusses our experiments with KDD'99 data. We summarize our conclusions in section 4.

2 SVDD-based Anomaly Detection

SVDD [8] is an unsupervised support vector machine algorithm for outlier detection. The goal of SVDD is to distinguish one class of data, called target data, from the rest of the feature space. To do this, SVDD learns an optimal hypersphere around target data after mapping the whole dataset to high dimensional feature space. The hypersphere as descriptive model for target data is used to classify data into target data or non-target data (also be called outliers). For SVDD-based anomaly detection we take normal data as target class and all kind of known and unknown attacks as outliers.

2.1 SVDD

Let $\{x_i\} \subseteq \chi$ be a training dataset of N data points, with $\chi \subseteq R^d$ Using a nonlinear transformation Φ from χ to some high dimensional feature space, we

search for the optimal enclosing hypersphere that is as small as possible while at the same time, including most of the training data. This can be formulated as the following optimization problem:

$$\min_{R,\xi,a} \quad R^2 + \frac{1}{vN}\sum_i \xi_i \tag{1}$$

$$\text{subject to} \quad \| \Phi(x_i) - a \|^2 \leq R^2 + \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, N \ ,$$

where a is the center of the hypersphere and R is its radius. Parameter v controls the tradeoff between the radius of hypershpere and the number of points that it contains. It is expected that if R and a solve this problem, the decision function $f(x) = sgn(R^2 - \| \Phi(x) - a \|^2)$ is determined by location of x in the feature space. To solve this problem we introduce the Lagrangian:

$$L = R^2 - \sum_i (R^2 + \xi_i - \| \Phi(x_i) - a \|^2)\alpha_i - \sum_i \xi_i \beta_i + \frac{1}{vN}\sum_i \xi_i \ . \tag{2}$$

Setting to zeros the derivative of L with respect to R, a and ξ_i, leads to

$$a = \sum_i \alpha_i \Phi(x_i), \quad \alpha = \frac{1}{vN} - \beta_i \leq \frac{1}{vN}, \quad \sum_i \alpha_i = 1 \ . \tag{3}$$

We then turn the Lagrangian into the Wolfe dual form with kernel function:

$$\min_{\alpha} \quad \sum_{i,j} \alpha_i \alpha_j K(x_i, x_j) - \sum_i \alpha_i K(x_i, x_i) \tag{4}$$

$$\text{subject to} \quad \sum_i \alpha_i = 1, \quad 0 < \alpha_i < \frac{1}{vN}, \quad i = 1, \ldots, N \ .$$

Throughout this paper we use the Gaussian kernel: $K(x_i, x_j) = exp(-q \| x_i - x_j \|^2)$, with width parameter q. The optimal α 's can be obtained after the dual problem is solved. Few special points with $0 < \alpha_i < 1/vN$ just lie on the surface of hypershere and are called *support vectors* . The first equation of (3) means that a can be expressed as the linear combination of $\Phi(x)$, and then R can be computed from any *support vector* x_k:

$$R^2 = \| \Phi(x_k) - a \|^2 = K(x_k, x_k) - 2\sum_i \alpha_i K(x_i, x_k) + \sum_{i,j} (i,j)\alpha_i \alpha_j K(x_i, x_j). \tag{5}$$

2.2 Change Detection Algorithm

The main idea of our change detection algorithm comes from the change point detection theory. The objective of change point detection is to determine if the observed time series is statistically homogeneous, and if not, to find the point in time when the change happens [9]. In our application, real time data from

sensors are processed into multi-dimensional time series. Compared with traditional change detection algorithm Cumulative Sum (CUSUM), our SVDD-based algorithm could be easily applied to multi-dimensional series.

The idea of SVDD-based change detection is simple. SVDD always try to find an optimal hypersphere for the target class, which is the great majority in the training data. Thus the region of the hypersphere is a representative of the probability density function that generates the target class. Hence, comparing the geometries and location of hyperspheres has the equal effect with comparing training data that the hypersheres build on.

Fig.1 demonstrates the change detection algorithm. Two adjoining sliding windows with same size m are placed on the series to produce adjoining subset of data flow. The two windows are moving forward with fixed increment step simultaneously. At time t, subset $W_1 = \{x_{t-m}, \ldots, x_{t-1}\}$ and $W_2 = \{x_t, \ldots, x_{t+m-1}\}$ are obtained by the two windows. If we use them as training data to build SVDD models independently, we get hypersphere S_1 defined by center a_1 and radius R_1 for W_1 and hypersphere S_2 defined by center a_2 and radius R_2 for W_2. A unexpected change at time t, which means a different distribution of data after t, may result in different location and geometries of S_1 and S_2. We use a change detection index $I(t)$ to reflect the dissimilarity between S_1 and S_2:

$$I(t) = \| a_1 - a_2 \| / (R_1 + R_2) \ . \tag{6}$$

According to 3.1, a_1, a_2, and $\| a_1 - a_2 \|$ can be computed:

$$a_1 = \sum_i \alpha_{1i} \cdot \Phi(x_{1i}), \qquad a_2 = \sum_j \alpha_{2j} \cdot \Phi(x_{2j}) \ ,$$

$$\| a_1 - a_2 \|^2 = \sum_{i,j} \alpha_{1i}\alpha_{1j}K(x_{1i}, x_{1j}) + \sum_{i,j} \alpha_{1i}\alpha_{1j}K(x_{1i}, x_{1j}) - 2\sum_{i,j} \alpha_{1i}\alpha_{2j}K(x_{1i}, x_{2j}).$$

Radii of the hyperspheres can also be computed from their *support vectors* according to (5). It can be found that, although $I(t)$ is defined in the feature space, it can be computed in the input space using the kernel function.

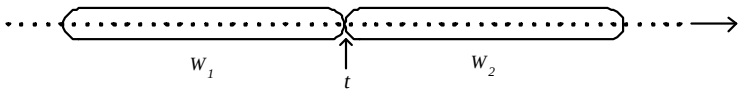

Fig. 1. Data series and sliding windows at time t. The right arrow indicates the direction of data generation.

With the continual generation of input data, the two windows are moving simultaneously with a fixed increment w that is predefined and $I(t)$ is computed every time. We then get a index curve of $I(t)$, and abrupt changes are easily detected whenever the index $I(t)$ peaks or is over a threshold λ.

There are two parameters, w and m, and a threshold λ which need to be considered. Window size m is selected based on several factors. It should not be too small. Otherwise, it can't reflect the data distribution, and will get $I(t)$ unsteady even for purely normal data flow. Nonetheless a too large m is also infeasible and unnecessary because it will increase the computing complexity. We are not able to find a universal value of m for any application, but we can find a proper value for our application by testing different m in normal data flow until getting steady change index values with a little variance. The moving increment, w, could range from 1 to m. This depends on the acceptable degree of detection delay. The index $I(t)$ measures the extent of change. In our application, we assume sudden a burst of large-scale intensive attacks will cause abrupt changes in data flow while concept drift raises mild and gradual changes in data flow. The threshold λ is used to detect abrupt change. If one $I(t)$ in index curve goes above λ, it indicates an ongoing intensive attack.

2.3 Adaptive Anomaly Detection Framework

Based on SVDD algorithm and change detection algorithm, we design an adaptive anomaly detection framework, which consists of four main components: preprocessor, change detector, model generator, and anomaly detector. The preprocessor transforms the raw network packets from sensors into formatted data, and then sends these data to the anomaly detector and the change detector. The anomaly detector uses a SVDD model to classify normal and intrusive data and raises alarm for ongoing intrusion. The change detector uses change detect algorithm to detect the intensive attacks and prepares training data for the model generator. The training data are stored in database and they are sent to the model generator when model update condition is triggered. The model generator learns a new model with new training data, and feeds model to anomaly detector periodically.

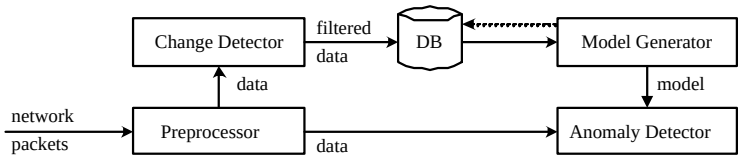

Fig. 2. Adaptive Anomaly Detection Framework

3 Experiment

We conducted experiments on KDD'99 dataset [10] , which is prepared for network intrusion detection. In the dataset, the network traffic data are connection-based. Each connection instance, described by 7 symbolic attributes and 34

continuous attributes, corresponds to a TCP/IP connection. The symbolic attributes must be transformed into numeric attribute to adapt to the SVDD algorithm. And attributes scaling is needed in order to ensure that the effect of some attributes is not dwarfed by others that have larger scales. The detail of these data preprocessing methods is described in our previous paper [11].

Experiment 1 (Exp1) is designed to evaluate the change detection algorithm for detecting intensive attacks. We take SYN flood DoS attack as an example. KDD'99 provides a typical 10 percent subset consisting of 494,020 instances, in which most instances are attacks. We reserve all of its 97,277 normal instances and filter most attacks to get a new set $C1$. In $C1$, 5 SYN flood attacks are reserved, which include more than 100,000 instances. Besides SYN flood attacks, all other kinds of at-tacks are less than 900 in $C1$.

We first illustrate how the change index can reflect the influence of SYN flood attacks. Fig. 3 displays the change index values obtained on $C1$ data flow, where the sliding windows size m is 3,000 and the increment w for windows is 3,000. SVDD parameter v is 0.001 and Gaussian kernel parameter q is 0.02.

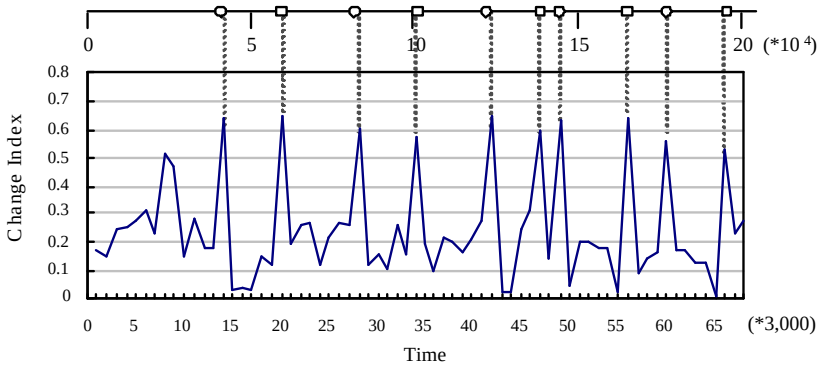

Fig. 3. Change index curve generated in $C1$. Top line gives the SYN flood attack schedule in $C1$. The circle symbol indicates the beginning of attack and square symbol indicates the ending. Corresponding change index is shown by bottom curve

In Fig. 3, when the threshold λ is 0.53, all changes caused by SYN flood attacks are correctly detected, with no false positive. It is natural that not only the starting of a SYN flood but also the withdrawing of the attack produce peaks in change index curve. These data falling in the two peaks should be rejected by the training dataset. In fact, the change detection algorithm not only can be used to prepare training dataset, but also can act as an intensive DoS attacks detector if we set a proper parameter, such as w. When detecting these kinds of DoS attacks, we are most concern with how to detect them as soon as possible so that we can take some response actions early to reduce the damage. With a 3,000 increment for windows, the average alarm delay for five

SYN flood attacks is 342 connections. This means that we become aware of an attack in its first 342 connections. We can use less increment step for window to get early change alarm. In the set $C1$, when w is 500, the average alarm delay is 105. Theoretically, a smaller w is good for less delay time, but in fact a very smaller w is unpractical because SVDD's efficiency problem though online version of SVDD [12] is employed.

Experiment 2 is designed to test our adaptive anomaly detection system. The experiment compares the performance of static method with adaptive learning strategy. On the basis of the Exp1, $C1$ is filtered and generate a new dataset $C2$ in which attack instances are about 1%.

In order to compare the adaptive manner with the static manner, first 20,000 normal records of $C2$ are extracted to get an initial training dataset. An initial model is build based on this initial training dataset. Exp2-1 is an experiment for adaptive manner which updates the model periodically. In this mode, a retrain period for model training and update is set. First the initial model is used, then at the end of every period, a new model is trained using data collected in this period and the old model is replaced with the new generated one. In Exp2-1, retrain period is set 20,000. Exp2-2 is a static manner experiment without updating the model. It just uses the initial model to detect the rest of $C2$ set, and the model remains unchanged during the detecting process.

Table 1. Results of Exp2-1 and Exp2-2

Experiment	Elapsed time (thousand of instances)			
	20	40	60	80 (all)
	False positive rate(%)			
Exp2-1	4.87	4.33	5.39	4.45
Exp2-2	4.87	6.43	9.52	13.43
	Detection rate(%)			
Exp2-1	96.33	92.66	88.76	89.27
Exp2-2	96.33	95.17	92.80	92.35

In initial dataset, the parameter v and q are selected through cross validation to obtain the minimum false positive rate. We set v 0.01 and q 0.5 when false positive rate is 1.06%. In Exp2-1, the two parameters are unchanged. Table 1 shows the detection rates and false positive rates for Exp2-1 and Exp2-2 over elapsed time, i.e. more instances are seen. The static model in Exp2-1 is able to detect 92.35% of the attacks in the dataset $C2$ at the end of all data. However, the false positive rate is increasing with time, and reaches 13.43% at the end, which indicates the influence of concept drift in $C2$. At the time, the adaptive manner (Exp2-2) continuously adapts to the concept drift and thus improves the false positive. Consequently, it generates significantly less false positive rate($< 5\%$) as well as a comparable detection rate with static model.

4 conclusion

Because of the limitation of application and the difficult of deployment of the previous adaptive system, in this paper, we present a new framework for adaptive anomaly detection based on SVDD. In order to implement the automatic collection of training data for model update, we design a change detection algorithm to find intensive attacks and to filter them from real time data. Then detection models are periodically regenerated with online collected training data. Our system significantly reduces human intervention as well as deployment costs. Results of experiments with the KDD'99 network data and preliminary analysis show that it can adapt to the network behavior changes while still detect attacks.

References

1. J.Mchugh. Intrusion and intrusion detection. International Journal of Information Security, 1(1): 14-35, 2001.
2. T.Lane, C.E.Brodley. Approaches to Online Learning and Concept Drift for User Identification in Computer Security. In Fourth International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, Aug. 1998, 259-263.
3. E. Eskin, M. Miller, Z.-D. Zhong, G. Yi, W.-A. Lee, and S. Stolfo. Adaptive model generation for intrusion detection systems. In Proceedings of the ACMCCS Workshop on Intrusion Detection and Prevention, Athens, Greece, 2000.
4. K. Yamanishi, J. Takeuchi, and G. Williams. On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA, Aug. 2000, 320-324.
5. M. Hossain and S. M. Bridges. A framework for an adaptive intrusion detection system with data mining.In Proceedings of the 13th Annual Canadian Information Technology Security Symposium, Ottawa, Canada, June 2001.
6. W. Fan. Cost-sensitive, scalable and adaptive learning using ensemble-based methods. Ph.D. dissertation, Columbia University, Feb. 2001.
7. E. Eskin, A. Arnold, M, Prerau, L. Portnoy and S. Stolfo. A Geometric Framework for Unsupervised Anomaly DetectionDetecting Intrusions in Unlabeled Data. In D. Barbara and S. Jajodia (editors), Applications of Data Mining in Computer Security, Kluwer, 2002.
8. D.M.J. Tax, R.P.W. Duin. Support vector domain description. Pattern Recognition Letters, 20(11-13): 1191-1199, 1999.
9. M. Basseville, I. V. Nikiforov. Detection of Abrupt Changes : Theory and Application. Prentice Hall, 1993.
10. http://kdd.ics.uci.edu/databases/kddcup99/task.html.
11. M.Luo, H.G Zhang , L.N.Wang, and J. Chen. A Research on Intrusion Detection Based on Unsupervised Clustering and Support Vector Machine. In Proceedings of Fifth International Conference on Information and Communications Security, Huhhot, China, Oct. 2003, 325-336.
12. D.M.J. Tax, P.Laskov. Online SVM Learning: From Classification to Data Description and Back. In Proceedings of IEEE International Workshop on Neural Networks for Signal Processing, Toulouse, France, September 17-19, 2003, 499-508.