Anomaly Intrusion Detection for Evolving Data Stream Based on Semi-supervised Learning

Yan Yu[1,*], Shanqing Guo[2,*], Shaohua Lan[1], and Tao Ban[3]

[1] School of Compuer Science and Technology,
Nanjing University of Science and Technology, 210094 Nanjing, P.R. China
{yuyan,lansh}@mail.njust.edu.cn
[2] School of Compuer Science and Technology, Shandong University,
250101 Jinan, P.R. China
guoshanqing@sdu.edu.cn
[3] Information Security Research Center, National Institute of Information and
Communications Technology, 184-8795 Tokyo, Japan
bantao@nict.go.jp

Abstract. In network environment, time-varying traffic patterns make the detection model not characterize the current traffic accurately. At the same time, the deficiency of training samples also degrades the detection accuracy. This paper proposes an anomaly detection algorithm for evolving data stream based on semi-supervised learning. The algorithm uses data stream model with attenuation to solve the problem of the change of traffic patterns, as while as extended labeled dataset generated from semi-supervised learning is used to train detection model. The experimental results manifest that the algorithm have better accuracy than those based on all historical data equivalently by forgetting historical data gracefully, as while as be suitable for the situation of deficiency of labeled data.

1 Introduction

Along with the development of computing and network technology, there has been a dramatic growth of interest in network security. In particular, Intrusion Detection Systems (IDS) have become important tools for ensuring network security. Intrusion detection is based on the assumption that intrusion activities are noticeably different from normal system activities and thus detectable. Earlier studies have utilized a rule-based approach for intrusion detection, but can not find the novel or unknown attacks[1,2]. As a result, various data mining techniques are used to detect attacks[3,4]. But there exist still some drawbacks, such as lack of labeled data, time-varying traffic patterns, low detection accuracy.

Traditional anomaly detection algorithms require a set of labeled data to train their intrusion detection models. But in network scenarios, there are a large number of unlabeled data but insufficient of labeled data since it will be expensive to generate.

* Corresponding author.

M. Köppen et al. (Eds.): ICONIP 2008, Part I, LNCS 5506, pp. 571–578, 2009.

Semi-supervised learning[5,6] combines labeled and unlabeled data during training to improve performance. In semi-supervised clustering, some labeled data are used with unlabeled data to obtain better clustering. It is known that applying semi-supervised learning to anomaly detection can improve the detection accuracy.

The consecutive arrival of network traffic demands the detection model having well scalability. Besides, time-varying traffic patterns demand that the detection model should adapt to the change of those patterns. But previous attempts to build intrusion detection systems have been based on mining static, previously stored network traffic[3]. Such approaches are not suitable for the temporal nature of network traffic, which is a data stream. A data stream is a continuum of potentially infinite data elements, in an ordered sequence of arrival. So, a data stream has necessarily a temporal dimension, and the underlying process that generates the data stream can change over time[7,8].

In the light, we introduce an data stream model and semi-supervised learning into intrusion detection area. In this paper, we propose an anomaly detection algorithm for evolving data stream based on semi-supervised learning, SSAD. SSAD algorithm utilizes attenuation rule to decrease the effect of historical data on detection result, which can help the algorithm to learn from current data that charaterize the traffic pattern more accurately. SSAD also use semi-supervised learning to extend labeled dataset as training dataset to do with the problem of lack of labeled data. The experimental result show that the algorithm not only improves the detection accuracy, but also be suitable for the situation of deficiency of labeled data.

The paper is organized as follows: section 2 introduces our semi-supervised learning approach to generate extended labeled dataset. Section 3 describes the data stream model for network traffic. Section 4 describes our SSAD algorithm. In section 5 we present the experimental results and analysis of SSAD algorithm in intrusion detection. The paper concludes with section 6.

2 Drawbacks of Existing Methods and Our Solutions

Existing intrusion detection approaches face with two problems mainly. Firstly, the traffic patterns change over time. Consequently, the detection model should charaterize the nature of traffic time-varying. In other words, data in different stage should have different influence on the detection model. Secondly, it is well known that it is extremely difficult to obtain a large number of labeled data as the training dataset of intrusion model in network scenarios, which would degrade the detection accuracy heavily. Therefore, we will analyze the problems and propose our solutions.

2.1 Influences of Data in Different Stages and Our Solution

A data stream can be viewed as an infinite process consisting of data which evolve with time. Among of all data, current data must have more influence than historical data at the respect of detecting attacks. Therefore, intrusion detection algorithms based on data stream model should process network data discriminatively according to their arriving time. The influence of historical data should fade out as time goes on.

At the same time, storing all historical data is unrealistic. So, we use history information vector to represent history information which is the extension of [9] on time dimension. Several related concepts are formally defined as follows:

Definition 1. (History Information Vector): History information vector in anomaly intrusion detection is defined as a clustering information set $\{\vec{HI}_p\}_{p=1}^{k}$, which represents d-dimensional set of datapoints $\{...,\vec{X}_{i-1},\vec{X}_i,\vec{X}_{i+1},...\}$, where $\vec{HI}_p$ denotes $(2d+1)$ tuple of the p-th cluster in history information. Namely, given d-dimensional datapoint set $\{\vec{X}_i \mid 0<i\le n\}$ in a cluster, there is $\vec{HI}_p=(\vec{S}_p,\vec{D}_p,n_p)$, where

$$\begin{aligned}\vec{S}_p&=\left[\sum_{i=1}^{n}x_i^{(1)},\sum_{i=1}^{n}x_i^{(2)},...,\sum_{i=1}^{n}x_i^{(d)}\right]\\ \vec{D}_p&=\left[\sum_{i=1}^{n}\left(x_i^{(1)}\right)^2,\sum_{i=1}^{n}\left(x_i^{(2)}\right)^2,...,\sum_{i=1}^{n}\left(x_i^{(d)}\right)^2\right]\end{aligned} \tag{1}$$

where n_p is the number of datapoints in cluster p.

Definition 2. (Information Attenuation): The influence of information on the result of intrusion detection will be fading out with time. This procedure is called information attenuation. Let $w_k^{(0)}$ denote the weight of cluster k in chunk, and α is attenuation coefficient. Then the weight will be changed to $w_k^{(t)}=e^{-\alpha t}w_k^{(0)}$ after period t, where α can be calculated by half-life ρ as follows: $e^{-\alpha\rho}=\frac{1}{2}$.

Attenuation Rule. Let $n_k^{(i)}$ denote the number of datapoints in cluster k which appear in the chunk at time t, and $c_k^{(t_0+t)}$ denotes one after period t. The attenuation rule can be defined as

$$\begin{cases}c_k^{(t_0)}=w_k^{(0)}n_k^{(t_0)}\\ c_k^{(t_0+t)}=e^{-\alpha}c_k^{(t_0+t-1)}+w_k^{(0)}n_k^{(t_0+t)}\end{cases} \tag{2}$$

Proof

Let t_0 be the initial time of algorithm. After every specific period, a data chunk will be generated. It is known that $w_k^{(t)}=e^{-\alpha t}w_k^{(0)}$ and $c_k^{(t_0)}=w_k^{(0)}n_k^{(t_0)}$. Then we have

$$\begin{aligned}c_k^{(t_0+t)}&=e^{-\alpha}c_k^{(t_0+t-1)}+w_k^{(0)}n_k^{(t_0+t)}\\ &=e^{-\alpha}(e^{-\alpha}c_k^{(t_0+t-2)}+w_k^{(0)}n_k^{(t_0+t-1)})+w_k^{(0)}n_k^{(t_0+t)}\\ &=e^{-\alpha}(e^{-\alpha}(e^{-\alpha}c_k^{(t_0+t-3)}+w_k^{(0)}n_k^{(t_0+t-2)})+w_k^{(0)}n_k^{(t_0+t-1)})+w_k^{(0)}n_k^{(t_0+t)}\\ &=\cdots\\ &=w_k^{(t)}n_k^{(t_0)}+w_k^{(t-1)}n_k^{(t_0+1)}+\cdots+w_k^{(0)}n_k^{(t_0+t)}\end{aligned}$$

According to the definition of information attenuation, the datapoint number of cluster k after period t should be

$$c_k^{(t_0+t)} = w_k^{(t)} n_k^{(t_0)} + w_k^{(t-1)} n_k^{(t_0+1)} + \cdots + w_k^{(0)} n_k^{(t_0+t)}$$

Attenuation rule is tenable.

2.2 Difficulty of Obtaining Large Number of Labeled Data and Our Solution

In network environment, it is extremely difficult to obtain a large number of labeled data. The deficiency of training samples degrades the detection accuracy. If we can extend the labeled data set to train detection model, we should build an accurate one. In clustering, an unlabeled dataset is partitioned into groups of similar examples, typically by optimizing an objective function. So, we consider to extend the labeled dataset using semi-supervised clustering to obtain more training samples.

There are both labeled and unlabeled data in original dataset. We can partition the data into different clusters by k-means algorithm. In k-means algorithm, objects in the same partition have high similarity. So, we can say that closer the distance between object and cluster center is, higher probability they have the same distribution with.

Therefore, we can extend labeled dataset by labeling some unlabeled data which is near to the labeled cluster center, according to the confidence $c\%$. The more the data are labeled, the more the training samples are. Accordingly, the noise introduced by labeling wrongly is more. So, the choice of confidence should balance between both of them. By this way, it is possible to obtain sufficient labeled data as training samples, which may improve the accuracy of learned detection model.

The concept of Extended Labeled Dataset is defined as following:

Definition 3. (Extended Labeled Dataset): Let $x \in S$ denote network data where $S = D_l \cup D_u$, $D_l = \{x_j, l_j | j = 1, \ldots, n\}$ denotes labeled dataset, $D_u = \{x_j | j = n+1, \ldots, n+m\}$ denotes unlabeled dataset with $m = |D_u| >> n$. Extending the label information of data in D_l to those in D_u according to the result of semi-supervised clustering. Then labeling the closest data to labeled cluster center based on confidence $c\%$, a new labeled dataset can be generated. The generated dataset is called extended labeled dataset on S, denoted by ED_l.

3 SSAD Algorithm

In SSAD algorithm, network data stream will be divided into chunks based on their arriving time. To every chunk of data and history information which has been process by attenuation, algorithm will apply semi-supervised clustering to extend labeled dataset which will be used as training dataset later.

Furthermore, SVM[10] has fast processing speed and good scalability, which have been proved to be suitable for intrusion detection[11]. Therefore, SVM algorithm is selected as classifier to build anomaly intrusion detection model. The main procedure of SSAD algorithm is described as follows.

Algorithm. SSAD

Input: data stream $X = \{D^{(t)} | t = 1, 2, \ldots\}$, where $D^{(t)} = D_l^{(t)} \cup D_u^{(t)}$ for t as the chunk sequence number, in which $D_l^{(t)} = \{(x_1^{(t)}, l_1), \ldots, (x_n^{(t)}, l_n)\}$ is labeled dataset, $D_u^{(t)} = \{x_{n+1}^{(t)}, \ldots, x_{n+m}^{(t)}\}$ is unlabeled dataset, confidence $c\%$, and half-life ρ.

Output: Intrusion detection model $N^{(t)}(x)$.

1. Initialize data structures, $t = 0$, $w_k^{(0)} = 1$;
2. Cluster labeled dataset $D_l^{(t)}$, if $x_i, x_j \in S_h \wedge S_h \subseteq D_l^{(t)}$, then $l_i = l_j$;
3. Cluster iteratively until convergence:
   3a. As to unlabeled dataset $D_u^{(t)}$, assign its datapoint to closest cluster;
   3b. As to history information vector $\{\overrightarrow{HI}_p\}_{p=1}^k$, assign it to closest cluster, distance can be calculated as $dist(\overrightarrow{HI}_p, \mu_h) = \sqrt{\frac{1}{n_p}[n_p \mu_h^2 + \sum_{i=1}^{n_p} x_i^2 - 2\mu_h (\sum_{i=1}^{n_p} x_i)]}$;
4. Label clusters as $label_i$ with $0 \le i \le \gamma$, if labeled data exist according the class of labeled data, where γ is known as the class number of labeled dataset $D_l^{(t)}$;
5. Determine the classe ξ_h of clusters, where $\xi_h = \begin{cases} label_j, & if \quad |label_i| \prec |label_j|, 1 \le i \wedge j \le \gamma \wedge i \ne j \\ new\_attack, & S_h \cap D_l = \Phi \end{cases}$, then distributing cluster class ξ_h to every member in the cluster;
6. Select $c\%$ data which are closest to their cluster center in each cluster, and put them into extended labeled dataset $ED_l^{(t)}$, where $c\%$ is the confidence;
7. Train SVM by $ED_l^{(t)}$, then generate intrusion detection model $N^{(t)}(x)$;
8. Calculate history information vector of current chunk as $\{\overrightarrow{HI}_p\}_{p=1}^k$ and its attenuation as $c_h^{(t+1)} \leftarrow e^{-\alpha} c_h^{(t)} + w_h^{(0)} |S_h^{(t)}|$;
9. $t = t + 1$, then goto step 2.

4 Experimental Results and Analysis

4.1 Dataset and Performance Measures

In order to evaluate the performance of SSAD algorithm, the dataset is generated randomly from a dataset in KDD[12], which contains 492,000 records having 41 features each. This dataset has five different classes including Normal traffic, DoS, Probe, R2L and U2R. In the experiment, small labeled data and plenty of unlabed data are sampled randomly. All models are trained and tested with the same dataset.

4.2 Performance Measures

The effect of extended labeled dataset on intrusion detection is evaluated by precision, recall and F-value[13], which may be defined as follows: precision is defined as the number of correctly detected attacks divided by total number of true attacks. recall is

defined as the number of correctly detected attacks divided by total number of attacks. And F-value is defined as (2 × recall × precision) divided by (precision + recall).

The accuracy of SSAD algorithm is evaluated by detection rate(DR) and false positive rate(FPR), which may be defined as follows: DR is definded as the number of correctly detected attacks divided by total number of true attacks. And FPR is defined at the number of false alarms divided by total number of normal samples.

4.3 Experiment and Analysis on SSAD Algorithm

Firstly, we evaluate whether the extended labeled dataset in SSAD will improve the accuracy of intrusion detection. In the experiment, we use only the extended labeled dataset, but not data stream model and history information attenuation, to evaluate the detection accuracy. SVM algorithm is used as classifier. Here, we run SVM with extended labeled dataset, abbreviated as SVMw, and SVM without extended labeled dataset, abbreviated as SVMw/o, ten times on the same training and test dataset, respectively. Then the average results are calculated. Fig.1 reports the comparison of F-value between SVMw and SVMw/o.

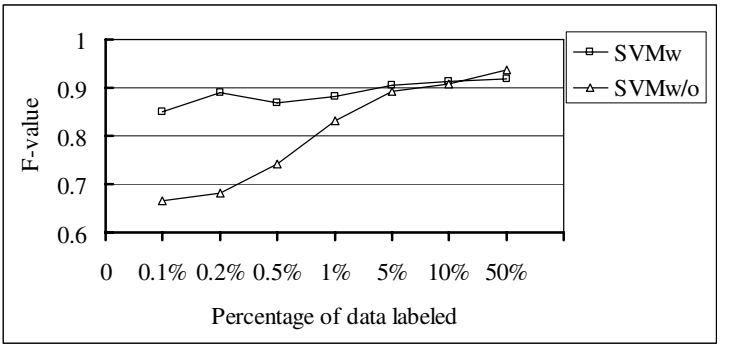

Fig. 1. Comparison between SVMw and SVMw/o with different proportions of labeled data

As represented in Fig. 1, detection accuracy of SVMw is better than SVMw/o apparently when the labeled data are scarce. With the increase of the number of labeled data, the curve of SVMw/o ascends sharply. When the proportion of labeled data reaches 50%, SVMw/o becomes better than SVMw. The reason is that the extended labeled dataset maybe consist of wrongly labeled data, which can degrade the accuracy of SVM. Therefore, it is considered that extended labeled dataset helps to improve the detection accuracy when the labeled data are scarce.

Secondly, we evaluate whether our SSAD algorithm can improve the detection accuracy in evolving data stream. In the experiment, we divide the dataset into 12 equal size of chunks according to their arriving time. Here, we assume that the data arrive with uniform speed. We apply SSAD with different half-lift ρ to each chunk where ρ equals 0.5, 1.0 and 5.0. None denotes that attenuation rule is not used. The DR and FPR of each chunk is showed in Fig. 2 and Fig.3, respectively.

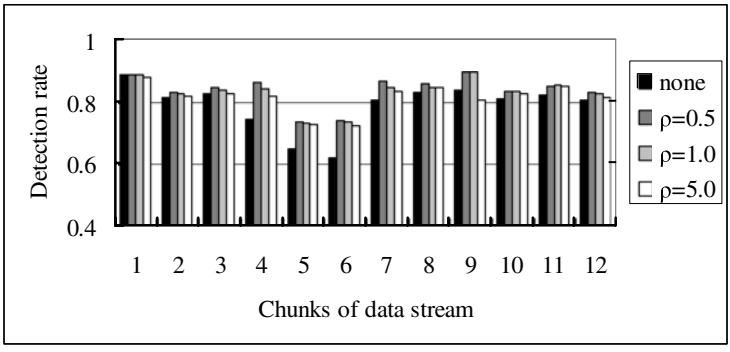

Fig. 2. Detection rate of SSAD algorithm for evolving data stream

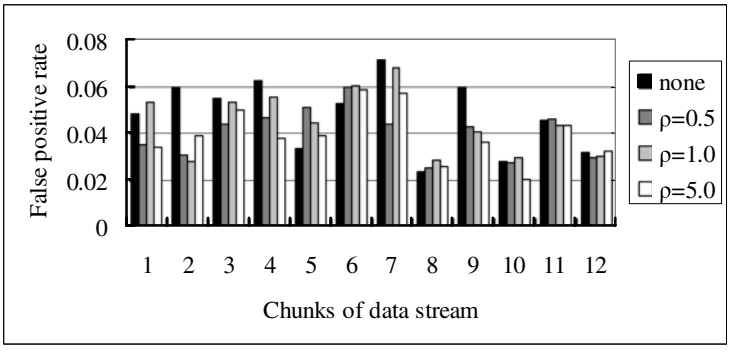

Fig. 3. False positive rate of SSAD algorithm for evolving data stream

From Fig. 2 and Fig.3, we can notice that the DRs without attenuation are the lowest among of all chunks under the condition of low FPR. That implies that the detection algorithm learned equivalently of all historical dataset can not adapt to the change of traffic pattern well. Therefore, it can be considered that forgetting historical data gracefully will help detection algorithm to adapt to the change of traffic pattern. Accordingly, detection accuracy will be better than that based on all historical data.

5 Conclusions

This paper proposes an anomaly detection algorithm for evolving data stream based on semi-supervised learning, SSAD. Aiming at the problem of deficiency of labeled data in network traffic, SSAD uses semi-supervised learning to extend labeled dataset. At the same time, data stream model is used to charaterize the time-varying traffic pattern, which help SSAD to adapt to the change of traffic pattern. The experimental results manifest that SSAD can achieve higher accuracy than that learned equivalently of all historical data, as while as be suitable for scarce labeled data.

References

1. Paxson, V.: Bro: A System for Detecting Network Intruders in Real-Time. Computer Networks 31(23-24), 2435–2463 (1999)
2. Porras, P.A., Neumann, P.G.: EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances. In: 9th National Computer Security Conference, pp. 353–365 (1997)
3. Burbeck, K., Nadjm-Tehrani, S.: ADWICE – Anomaly Detection with Real-Time Incremental Clustering. In: Park, C.-s., Chee, S. (eds.) ICISC 2004. LNCS, vol. 3506, pp. 407–424. Springer, Heidelberg (2005)
4. Kasabov, N.: Evolving Connectionist Systems: The Knowledge Engineering Approach. Springer, London (2007)
5. Basu, S., Bilenko, M., Mooney, R.J.: A Probabilistic Framework for Semi-Supervised Clustering. In: 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 59–68. ACM Press, New York (2004)
6. Wagstaff, K., Cardie, C., Rogers, S., Schroedl, S.: Constrained K-Means Clustering with Background Knowledge. In: 18th International Conference on Machine Learning, pp. 577–584. Morgan Kaufmann, San Francisco (2001)
7. Babcock, B., Babu, S., Datar, M., Motwani, R., Widom, J.: Models and Issues in Data Streams. In: 21st ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp. 1–16. ACM Press, Madison (2002)
8. Aggarwal, C.C.: On Change Diagnosis in Evolving Data Streams. IEEE Trans. Knowl. Data Eng. 17(5), 587–600 (2005)
9. Zhang, T., Ramakrishnan, R., Livny, M.: BIRCH: An Efficient Data Clustering Method for Very Large Databases. In: 1996 ACM SIGMOD International Conference on Management of Data, pp. 103–114. ACM Press, Montreal (1996)
10. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, New York (1995)
11. Mukkamala, S., Sung, A.H., Abraham, A.: Intrusion Detection using An Ensemble of Intelligent Paradigms. J. Netw. Comput. Appl. 28(2), 167–182 (2005)
12. The UCI KDD Archive, `http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html`
13. Joshi, M., Agarwal, R., Kumar, V.: Predicting Rare Classes: Can Boosting Make Any Weak Learner Strong? In: 8th ACM Conference ACM SiGKDD International Conference on Knowledge Discovery and Data Mining, pp. 297–306. ACM Press, Edmonton (2002)