One Class Support Vector Machine for Anomaly Detection in the Communication Network Performance Data

RUI ZHANG, SHAOYAN ZHANG, SETHURAMAN MUTHURAMAN, JIANMIN JIANG
School of Informatics
University of Bradford
Richmond Road, Bradford, BD7 1DP
UNITED KINGDOM

Abstract: - Anomaly detection refers to the automatic identification of the abnormal behaviors from a large amount of normal data. Anomaly detection is more and more required in the communication network due to the increasing number of the unauthorized activities occurring in the network. This paper presents a method based on one class support vector machine (OCSVM) to detect the network anomalies. The communication network performance data are used for the investigation and the raw data are firstly preprocessed in order to produce the vector sets required by the OCSVM algorithm. The training data are used to train the OCSVM anomaly detector, and the trained detector is applied on the test data to detect the anomalies. In addition, the results are compared with the results obtained from the rule-based system which is currently used in the communication network. The algorithm shows the promising performance on the network anomaly detection and provides a great reduction on the volume of the alarms than the rule-based system.

Key-Words: - anomaly detection, artificial intelligence, communication network, one class support vector machine, performance management, time series

1 Introduction

As more and more unauthorized activities occur in the communication performance data, intelligent management of communication data and automatic detection of network anomalies are greatly demanded. Network performance data include qualitative data, also known as Key Performance Indicators (KPIs), and quantitative data. These data are used to characterize the network behavior and therefore used for network anomaly detection. In general, if a value in a data set falls outside of the so-called "normal patterns", which represent normal and faultless network behavior, an alarm is generated. A particular network problem can cause many different alarms because one anomaly can be tied to many different child root causes, and the problems with network failures can also result in the drop or rise of many different performance management (PM) indicators. In the communication systems, the alarms are generated from analyzing KPIs and quantitative data.. Metrics are calculated for any logical and physical network component for which raw performance data when counters are available. Definition of individual metrics pertains to the network domain (e.g. BSS, GSN, and MSC). For an example a network domain contains many BSS's elements. The KPI for the network domain is determined by combining many BSS quantitative data. There are 10 -100 KPIs types per element network domain. KPIs are measured in percent (A/B) or quantity only (numerator A or denominator B). Say A and B are specific counters in the network nodes. KPIs are also derived by compound of many quantitative data.

Every data set for every element in the network has its own typical trend. Within the different type of elements, it has different typical values for the KPIs. Anything that is suddenly different from the normal or typical value is abnormal and that would indicate an alarm on the PM KPIs. With quantitative data, alarms are normally generated when the trend suddenly sinks. In some situations alarms are also generated on sudden rising of data values. As a result of poor setting of the threshold for different indicators or due to very critical situations, the PM operator screen floods with alarms. Therefore, artificial intelligence (AI) based detectors are expected to learn the behavior of the indicators and to drastically reduce the volume of alarms presented to the administrator.

Researchers have reported different methods

for the network anomaly detection, which include the application of time series analysis [1], wavelet analysis [2], Neural Network [3, 4] and clustering [5]. In this paper, a method based on one class support vector machine (OCSVM) is proposed to generate alarms and the results are also compared with those provided by the rule-based system. The method consists of two main steps: first is the detector training, the training data set is used to get the OCSVM anomaly detector which is capable of learning the nominal profile of the data, and the second step is to generate the alarms in the performance data using the anomaly detector.

The rest of this paper is organized as follows: a brief introduction of the OCSVM theory is given in Section 2. The data source and the proposed algorithm are presented in Section 3. Experiments are carried out on the network traffic data of four consecutive weeks. The anomaly detection results are given in Section 4 and a comparison is made between the results of our algorithm and that of the rule-based system. Conclusions are finally made in Section 5.

2 Methodology

In this section, some basic concepts of the OCSVM will be introduced. The support vector machine (SVM) algorithm [6] as it is usually constructed is essentially a two-class algorithm. Scholkopf et al. [7] proposed a method of adapting SVM to one class classification problem. The OCSVM [8, 9, 10] can be considered as a regular two-class SVM where all the training data lies in the first class and the origin is the only member of the second class. The basic idea of the OCSVM is to map the input data into a high dimensional feature space using an appropriate kernel function and constructs a decision function to best separate one class data from the second class data with the maximum margin. In this paper, both the origin and the data that "close enough" to the origin belong to the second class and they are considered as the anomalies. Fig.1 gives the geometry interpretation of the OCSVM [11].

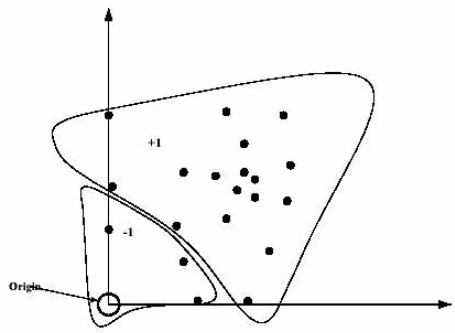

Fig. 1. One class SVM classifier.

More specifically, given a training data set without any class information, $x_i \in R^n, i = 1,2,...,l$, i is the number of data points in the training set, R^n is the input space and n is the dimension of the input space. $\Phi(x)$ is a map function that transforms x from the input space to the feature space F. A hyper-plane or linear decision function *f(x)* in the feature space F is constructed as

$$f(x) = w^T \Phi(x) - \rho \quad (1)$$

to separate as many as possible of the mapped vectors $\{\Phi(x_i), i = 1,2,...,l\}$ from the origin, w is the norm perpendicular to the hyper-plane and ρ is the bias of the hyper-plane. In order to solve w and ρ, it needs to solve the following optimization problem,

$$\min_{w,\xi,\rho} \frac{1}{2} w^T w + \frac{1}{vl} \sum_{i=1}^{l} \xi_i - \rho \quad (2)$$

subject to $w^T \Phi(x) \geq -\rho - \xi_i$, $\xi_i \geq 0, i = 1,2,...,l$

where ξ_i are slack variables that are penalized in the objective function and $v \in (0,1)$, which is the parameter that controls a trade off between maximizing the distance of the hyper-plane from the origin and the number of data points contained by the hyper-plane, i.e., when v is small, fewer data fall on the same side of the hyper-plane as the origin in the feature space F.

In order to solve (2), Largrangian multiplier α_i is introduced to each xi and the dual problem of the optimization problem of (2) can be obtained. Solving the dual problem leads to

$$w = \sum_{i=1}^{l} \alpha_i \Phi(x_i) \quad (3)$$

where $0 \leq \alpha_i \leq \frac{1}{vl}$. Accordingly the decision function f(x) becomes a nonlinear function as

$$f(x) = \sum_{i=1}^{l} \alpha_i K(x_i, x) - \rho \quad (4)$$

where $K(x_i, x) = \Phi(x_i)^T \Phi(x)$ [12], which is a kernel function in the input space. There are many admissible choices for kernel function depending on the experience and experiment. Keerthi and Lin [13] reported that Radial Basic Function (RBF), as shown in (5) is the most widely used kernel in SVM, and RBF kernel is adopted in our experiments. For any x, if f(x) is negative, x is detected as an anomaly, otherwise x is normal.

$$K(x_i, x) = e^{(-\|x_1 - x_2\|^2 / 2\sigma^2)} \quad (5)$$

3 Experiment

3.1 Data Set

In the communication system, the qualitative data have more complex features than the quantitative data. As most of the qualitative data have the same values, a threshold can be usually set to detect the anomalies, i.e., any value lower than the threshold can be considered as abnormal. However, it is difficult to choose a threshold for the quantitative data because of its non linear feature with varying values. We will focus on the analysis of quantitative data in this paper and the network traffic data are used for the investigation.

Due to the changes of network traffic on different days, the training data and the test data are needed to be obtained from the same day of the week so that the normal profiles learned from the training data can be used for the anomaly detection on the test data. The whole data set includes four recordings, which are recorded on the same day of four consecutive weeks. Each recording is made every five minutes during 24 hours (from 00:00am to 23:55pm) and thus 288 data points are obtained throughout one day for each recording. Three recordings of Weeks 1, 2 and 3 are used for training the OCSVM anomaly detector and the remaining one recording of Week 4 are used as the test data to evaluate the performance of the algorithm. Fig. 2 – 5 give the plots of the four weeks data. It can be seen from these plots that all these recordings have the similar trend, but at some data points the data value suddenly increases or drops. Such data points are the abnormal points that can be easily noticed by human visual detection and some points that can not be visually found are also possibly anomalies. The anomaly detector needs to find out both the visible and invisible abnormal data points.

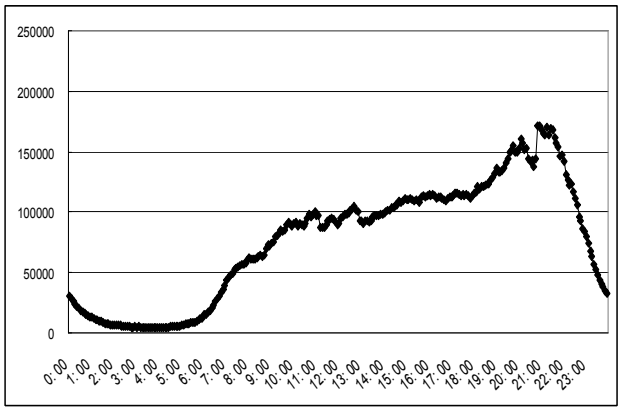

Fig. 2. Network traffic data of Week 1 with 288 data points throughout one day.

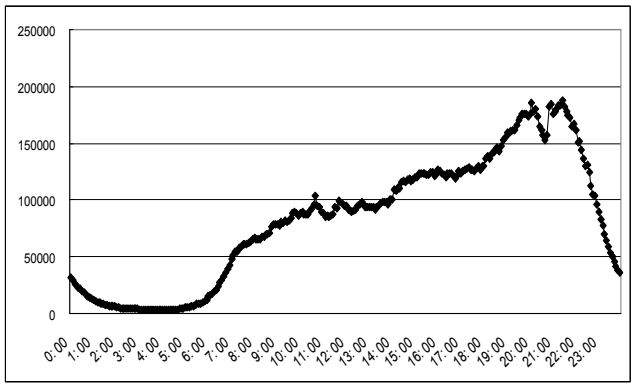

Fig. 3. Network traffic data of Week 2 with 288 data points throughout one day.

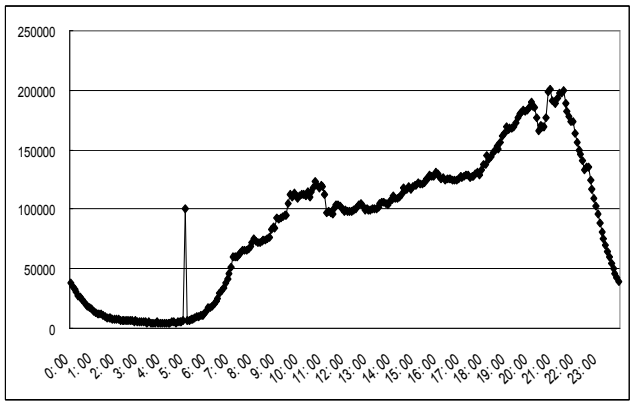

Fig. 4. Network traffic data of Week 3 with 288 data points throughout one day.

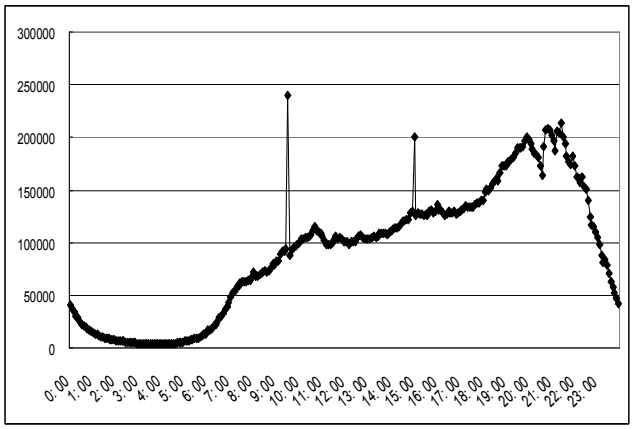

Fig. 5. Network traffic data of Week 4 with 288 data points throughout one day.

.

3.2 OCSVM for Anomaly Detection

3.2.1 Data Preprocessing

As discussed previously, the OCSVM can only be applied to a set of vectors, however our data are time series and required to be transfered into vector sets.

Given a time series of x(t), t =1,2,…..288, x(t) is converted into its feature space Q, where $Q \in R^n$, and the dimension of Q is set to be n = 2. In OCSCV, as the detection results will be heavily biased to the time series points with either extremely large values, the set of vectors are processed to make sure that most of the values are close to the origin, and thus for the quantitative data as in Fig. 1, the vector set is the difference between each pair of the adjacent data points divided by a certain number. The number is chosen based on the value of the time series and in our case this number is selected to be 1000 in order to keep the vectors close enough to the origin. The time series x(t) is preprocessed to a vector set using the following operations:

$X_2(t) = [x(t)\ \ x(t)]$, t=1,

$X_2(t) = [x(t)-x(t-1)\ \ x(t)-x(t-1)]/1000$, t=2,3…,288.

3.2.2 Detector Training and Anomaly Detection

As the RBF kernel is selected for the OCSVM algorithm, two parameters μ and σ need to be set before carrying out the training. The choice of the parameters depends on the requirements of the research problem. In our experiments, the two parameters of the algorithm are set as follows:

1) OCSVM parameter μ=0.01

2) RBF kernel function

$$K(x_i, x) = e^{(-\|x_1 - x_2\|^2 / 100)}$$

The purpose of OCSVM training is to create the OCSVM anomaly detectors that are capable of recognizing network data profile. Firstly, both the training set and the test set are preprocessed to obtain the training and test vector sets. The training vector set is then used to train the OCSVM detector and the OCSVM detector is then applied on the test vector set. If the return value for the decision function *f(x)* is negative, an anomaly is detected, otherwise it is normal.

4 Experimental Results

The experimental results are presented in this section to demonstrate the performance of the algorithm. The results are also compared with the results obtained from the rule-based system which is currently used in the communication system.

4.1 Anomaly Detection Results

The anomaly detection results of the test data are shown in Fig. 6. The original value of each data point of Week 4 is plotted on the lower line and its return value from the decision function is displayed on the upper line accordingly. Ten of 288 data points in the test data are detected as the anomalies. It can be seen from the plot that the anomalies happen when there is a big drop or increase on the data and the results perfectly matches the human visual detection result. For all the normal data points, the return values are equal to 1 and for the anomaly data points, the values are below 0. Details of the return values for these anomaly data points are given in Table 1. As the vector set is calculated as the difference between each pair of the adjacent data points, once the value of a data point that further derives from the previous value, a smaller negative value is obtained from the decision function. In other words, these returned values reflect the degree the data value derives from the normal data and therefore they can be used to determine the severity of the anomalies. The severity of an anomaly indicates the order in which users should handle that event relative to anomaly of other severities and this allows the users to prioritize their work, which is very meaningful in the real network application.

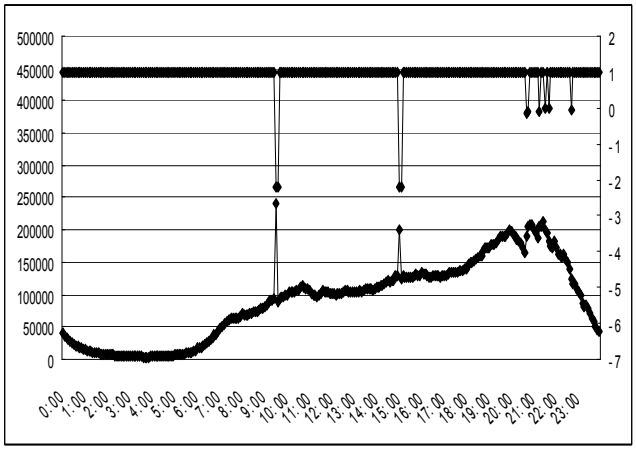

Fig. 6. Anomaly detection results of the test data.

Table 1.
Return values of the abnormal data points

Time	Original Data value	Return value
9:30am	240,000	-2.1956
9:35am	87,598	-2.1956
15:00pm	200,000	-2.1956
15:05pm	125,223	-2.1939
20:40pm	190,898	-0.1538
20:45pm	206,720	-0.0868
21:15pm	206,135	-0.1055
21:30pm	199,895	-0.0315
21:40pm	182,277	-0.0046
22:40pm	124,650	-0.0710

4.2 Results Comparison with the Rule-based System

The rule-based system is normally adopted to generate alarms in the current communication network. The main scheme of such system is to compare the new recordings with the so-called "normal pattern". The normal pattern is usually calculated using the median values of the recordings from several consecutive weeks and the median values of the differences between every two consecutive weeks. The anomaly detection rule of the system is that if the value of any certain data point is lower than that of the normal pattern, an alarm will be generated. The same training data including the recordings of Weeks 1, 2 and 3 are applied in the rule-based system and the normal pattern obtained from the training data is shown in Fig. 7. The data of Week 4 are again used for testing and the detection results of the rule-based system are shown in Fig. 8. In Fig. 8, the original value of each data point of Week 4 is plotted on the lower line and if its corresponding value on the upper line is -1, it indicates an alarm is generated at this data point and the value of 1 on the upper line indicates there is no alarm generated. 124 anomalies are detected while comparing Week 4 data with the normal pattern. This number of alarms provided by the rule-based system is extremely bigger than the alarms we get with our methods.

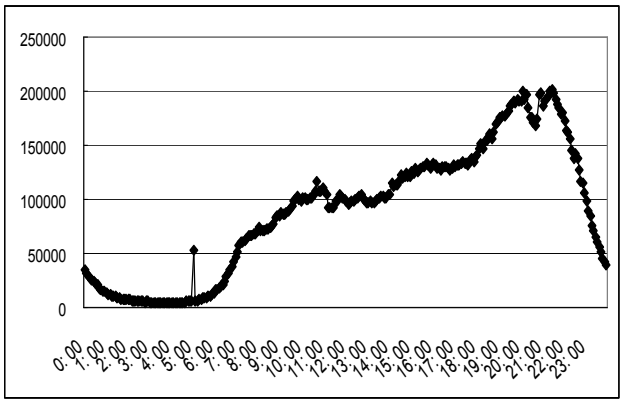

Fig. 7. Normal patterns of the traffic data using the training set.

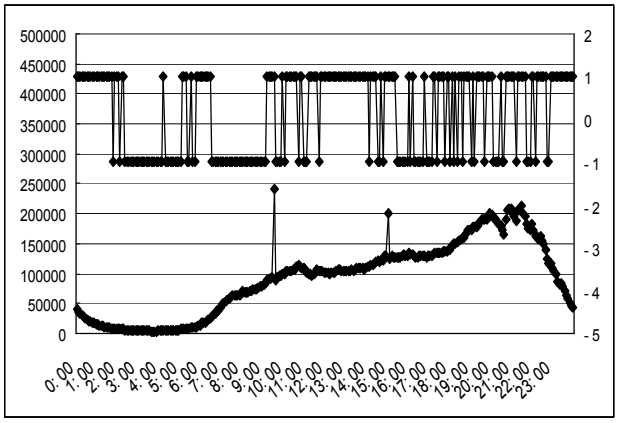

Fig. 8. Anomaly detection results of the rule-based system.

As seen from Fig. 6, the normal pattern contains a sudden increasing data point at 4:00am with the value being 53,404. This is caused by the big increase that happened at the same time point in Week 3. By looking at the data values of Week 1 and Week 2, this big value in Fig. 6. should be abnormal rather than normal. While this normal pattern is used for detection, a false alarm will be generated, e.g., the value of Week 4 at 4:00am is 6,637 and it is detected as an alarm as it is smaller than 53,404, however it should be actually a normal value. Therefore, the selection of the training set is very critical in such rule-based system and the normal data are required as training data to produce the normal

patterns, nevertheless such normal data are not easily to be attained in a real communication system. In the OCSVM approach, there is no such strict requirement for the selection of training data, the anomaly detector can be trained even with the abnormal data. Additionally, the rule-based system considers any drop of the value as an alarm while comparing with the normal pattern and this can also produce many false alarms. For instance, the data point of Week 4 at 9:30am has a value of 240,000 which is larger than the value of the normal pattern at this point, and thus it is detected as a normal point by the rule-based system. However this data point is determined as abnormal by human visual detection, which is consistent with the result attained from our approach. Among all these 124 alarms provided by the rule-based system, most of them are proved to be false alarms. Our approach only provides the most major alarms and gives a significant reduction on the amount of the alarms reported to the administrator.

5 Conclusion

Anomaly detection refers to automatic identification of abnormal events embedding in a large amount of data and it has been acquiring increasing attention because of its huge potential for application. This paper proposed a network anomaly detection method based on the OCSVM. This algorithm achieves the automated detection of the anomalies and also provides a great reduction on the number of alarms compared with the current rule-based system.

Experiments are carried out on communication network traffic data of four weeks and the results show the promising performance of this approach. By comparing the reults of our algorithm with the current rule-based system, it shows that this algorithm is more effective than the current rule-based systems. The rule-based system requires normal data to get a good normal pattern for detecting the anomalies and the detection rule is not flexible enough. However, the proposed algorithm is able to train the anomaly detector, which can learn the normal behavior with both the normal and abnormal data, and the anomaly detector can also detect new events which have never encountered during the training process. Moreover, this algorithm greatly reduces the volume of the alarms, e.g., the number of alarms is reduced from 124 to ten in our case. The effictive alarm detection and the reduction of the amount of the alarms provide great advantages in the real communication network application.

References:

[1] Q. Wu and Z. Shao, Network anomaly detection using time series analysis, Proceedings of the Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services, vol. 23-28, 2005, pp. 42-47.

[2] C. Shahaibi, X.Tian and W. Zhao, TSA-tree: wavelet-based approach to improve the efficiency of multi-level surprise and trend queries on time-series data, Proceedings of 12th International Conference on Scientific and Statistical Database Management, Germany, 2002, pp. 1-14.

[3] S. Muthuraman and J. Jiang, Anomaly detection in telecommunication network performance data, Proceedings of the International Conference on Artificial Intelligence, Las Vegas, Nevada, USA, June, 2007.

[4] S.-J. Han and S.-B. Cho, Evolutionary neural networks for anomaly detection based on the behavior of a program, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 36, no. 3, 2006.

[5] S. Salvador, P. Chan and J. Brodie, Learning states and rules for detection anomalies in time series, Applied Intelligence, vol. 23, no. 3, 2005, pp. 241-255.

[6] C. Burges, A tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery, vol. 2, no. 2, 1998, pp.121-167.

[7] 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. 3, no. 7, 2001, pp. 1443-1472.

[8] J. Ma and S. Perkins, Time-series novelty detection using one-class support vector machines, Proceedings of the International Joint Conference on Neural Networks, July, 2003, pp. 1741-1745.

[9] K. Li, H. Huang, S. Tian and W. Xu, Improving one-class SVM for anomaly detection, Proceedings of the Second International Conference on Machine Learning and Cybernetics, Xi'an, 2003, pp. 3077-3081.

[10] Katherine A Heller, Krysta M Svore, Angelos D. Keromytis, and Salvatore J. Stolfo. One Class Support Vector Machines for Detecting Anomalous Window Registry Accesses, Proceedings of the 3rd IEEE Conference Data Mining Workshop on Data Mining for Computer Security, Florida, 2003.

[11] L.M. Manevitz and M. Yousef, One-Class SVMs for document classification, Journal of Machine Learning Research, vol. 2, 2001, pp. 139-154.

[12] V.N. Vapnik, An overview of statistical learning theory, IEEE Trans. On Neural Networks, vol. 10, issue 5, 1999, pp. 988-999.

[13] S.S. Keerthi and C.J. Lin, Asymptotic behaviors of support vector machines with Gaussian Kernel, Neural Computation, vol. 15, no. 7, 2003, pp. 1667-1689.