Feasibility of One-Class-SVM for Anomaly Detection in Telecommunication Network

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

Abstract: - The growing number of unauthorized activities and various trends of networking technologies in telecommunication network have added heavy burdens to telecommunication performance management (PM) system. One-class-support vector machine (OCSVM) is introduced in this paper, to automatically detect network anomalies. Real telecommunication performance data are employed in this paper to investigate the feasibility of OCSVM for anomaly detection. Experiments with small and large data sets demonstrate that OCSVM can not only detect the anomalies correctly, but also fast in a short time. The promising performances show that OCSVM is efficiently enough to meet with the anomaly detection task in telecommunication network.

Key-Words: Telecommunication, Anomaly detection, Performance management, OCSVM

1 Introduction

Telecommunication networks are becoming more and more complicated, which brings about big operational problems for telecommunication operators. In order to indicate the performance of the network, network elements generate performance data. The data for each element has its own trend and typical values, which can characterize the network behavior and therefore is used for network anomaly detection. The anomaly is the abnormal behavior from normal trend and anomaly detection identifies the anomaly activities [1].

The performance of anomaly detection relies on the following two parameters – accuracy and efficiency. Accuracy means the detection method and detect the anomalies correctly, and efficiency means that the detection has to be fast enough to investigate the performance data in a short given time.

Several anomaly detection techniques and algorithms have been reported. One method is to define the abnormal conditions [2], however, due to the difficulty of defining unknown behaviors, these algorithms are always inappropriate for applications. Many algorithms, such as probabilistic techniques [3], neural network [4, 5], support vector machines [6, 7], K-nearest neighbor (KNN) [8] and Hidden Markov model [9], treat the anomaly detection as binary classification problem, however, strictly speaking they are not anomaly detection algorithms, as they require knowing what kind of anomaly is expecting, furthermore, these algorithms may be sensitive to noise in the training samples. Segmentation and clustering algorithms [10, 11] do not need to know the signatures of the series, the shortages are that they always need parameters to specify a proper number of segmentation or clusters and the detection procedure has to shift from one state to another state. Negative selection algorithms [12, 13] are designed for one-class classification; however, these algorithms can potentially fail with the increasing diversity of normal set and the computation procedure may not be efficient enough to finish detecting a large amount of data in a short given time.

One-Class-SVM (OCSVM) was proposed to detect the outliers [14-17]. The idea of OCSVM is to map the data into a high dimensional Hilbert space, and then maximize the margin between the mapped data and the origin in Hilbert space. A trade-off parameter ν is introduced in the objective function, which permits a maximum of $\nu \times 100\ \%$ data (anomaly data) to stay at the same side with the origin. In other words, OCSVM can distinguish the anomalies from nominal data. The OCSVM overcomes the shortcomings of the traditional SVMs for coping with the one class data with noise and is supposed to possess good generation ability in outlier detection. In this paper, the OCSVM will be introduced to the telecommunication network PM system, and will be verified via real telecommunication performance data for anomaly detection.

The rest of this paper is organized as follows: Section 2 presents a brief introduction of the OCSVM. Section 3 introduces the PM anomaly detection system which includes the detector training module and anomaly detection module. Experiments are executed in Section 4 to see the feasibility of OCSVM for anomaly detection. Conclusion and future work are discussed in Section 5.

2 A Brief Formulation of the OCSVM

Considering a data set with $T = \{x_1, x_2, \cdots, x_l\}$, $x \in R^N$, the task is to find a function $f(x)$ that takes the value "+1" for most of the vectors in the data set (marked by stars in Fig. 1 for 2-dimensional case), and "-1" for the other very small part (marked by circles).

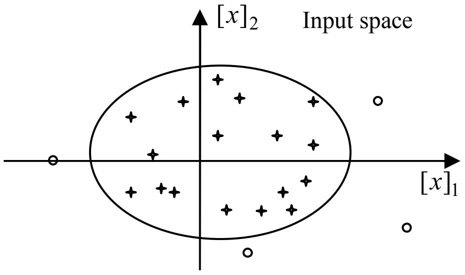

Fig. 1 – A data set in the input space

The strategies for the OCSVM are: first of all, map the input data into a Hilbert space H according to a mapping function $X = \phi(x)$, as demonstrated in Fig. 2, and then separate the data from the origin to its maximum margin and a hyper-plane *f(x)* is built up to mark the boundary of separation.

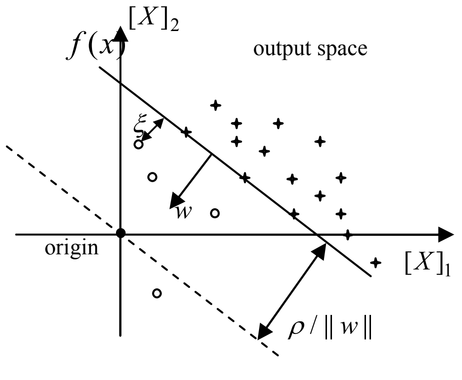

Fig. 2 – Data set in the output space after mapping

The key idea for the separation is that it doesn't really need all the data to be separated to the same side of the hyper-plane *f(x)*, on controversy, a small number of points can be lying on the other side of the hyper-plane. In order to allow this, slack variables are introduced to the objective function of support vector machine, and the OCSVM solves the following quadratic optimization problem:

$$\min_{w \in F} : \quad \tfrac{1}{2} \| w \|^2 + \frac{1}{\nu l} \sum_i \xi_i - \rho \tag{1}$$

$$\text{s.t.} \quad f(x) = w.\phi(x_i) - \rho \geq -\xi_i, \xi_i > 0, i = 1, \cdots, l \tag{2}$$

In Functions (1) and (2), w is the norm that perpendicular to the hyper-plane and ρ is the bias of the hyper-plane. ξ_i are slack variables acting as penalization in the objective function. $\nu \in (0,1)$ is the trade-off parameter to balance between the normal and anomaly data in the data set. Deriving its dual representations, the OCSVM is to solve the following problems:

1 Select the kernel function $K(x, x')$ and the trade-off parameter ν, construct and solve the following optimization problem to find the solution $\alpha^* = (\alpha_1^*, \cdots, \alpha_l^*)$:

$$\min_{\alpha} : \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j K(x_i, x_j) \tag{3}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq 1/(\nu l), i = 1, \cdots, l \tag{4}$$

$$\sum_{i=1}^{l} \alpha_i = 1 \tag{5}$$

where $K(x_i, x_j) = \phi(x_i).\phi(x_j)$ is called as kernel function and can be with various format.

2 Select any α^* with $0 < \alpha^* < 1/(\nu l)$ and calculate the bias $\rho = \sum_{i}^{l} \alpha_i^* K(x_i, x_j)$, the vectors which satisfied $0 < \alpha^* < 1/(\nu l)$ are called support vectors.

3 Integrate the decision function $f(x) = \sum_{i=1}^{N_{sv}} \alpha_i^* K(x_i, x) - \rho$, if $f(x) \geq 0$, return +1; otherwise, return the negative value. N_{sv} is the number of support vectors.

3 Anomaly Detection System for Telecommunication Data

3.1 Data Type and Anomalies in Telecommunication

Performance management in telecommunication addresses the problems of intelligently managing the performance data. One function of PM is to detect the anomalies and then generate alarms according to the detection results. There are two different types of PM data, qualitative data and quantitative data. Qualitative data, also known as key performance indicators (KPIs), measure the service quality. These indicator values are measured in percentage between zero and hundred. Instead of recording the percentage number, quantitative type data trace the traffic data at each service point in the network. Fig. 3 shows the normal performance quantity traces of a service for the same day (Thursday) in five weeks. The values of the quantitative data are logged at 15 minutes intervals throughout the day, and thus 96 raw values per day per recording. In the Fig., the obvious anomaly data are marked with circles. The objective of the PM system is to flag these anomalies occurred in the network.

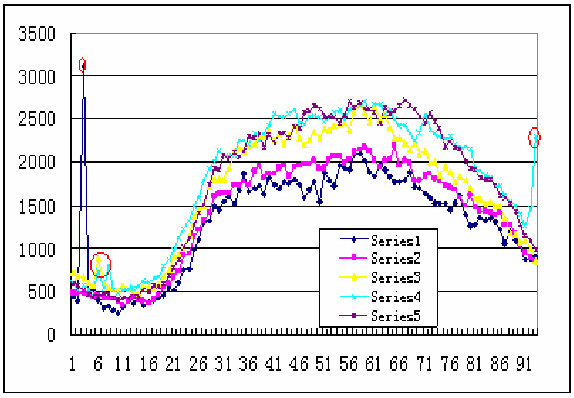

Fig. – 3 Quantitative data for the same day in five weeks

3.2 PM Anomaly Detection System

Anomaly detection system can be built in two steps: Firstly, the offline data is used to train the OCSVM and generate the model function $f(x)$; once testified, the model will be transferred to performance monitoring and management system to detect the anomalies in the online performance data.

As mentioned in step 3 of Section 2, the real negative values of the anomalies will be returned. These negative values can in fact reflect the degree of deviation of abnormal events, the lower the value, the more abnormal the event. According to different values, the detected anomalies can be clustered into three types, namely: severe, medium and minor alarms.

4 Experiments: Telecommunication Anomaly Detection Using OCSVM

In this part, experiments on real data will be carried out to check the feasibility of OCSVM for telecommunication anomaly detection. The experiments will be concentrated on two aspects: accuracy and efficiency. In our experiments, we focus on the analysis of the traffic data, which is one type of the quantitative data. The anomaly detection in traffic data (Fig. 3) seems is more difficult than that of percentage data, due to its nonlinear nature of the performance curve.

4.1 Experiment on a Small Data set

4.1.1 Data Pre-processing and Feature Extraction

This experiment is based on the data described in Fig. 3. The traffic data in the Fig. possesses a nonlinear nature of the performance curve, and it differs in the ranges of the values among different weeks. Due to its nonlinear nature, it is unable to use the data directly for training the OCSVM model. There is knowledge for telecommunication network that network failures cause the sinking or rising of different PM indicators, as a result, the first order gradients of the data set are generated to compose the new feature for training and testing. That is, a feature set $F = \{y_1, y_2, \cdots, y_l\}$ of a training set $T = \{x_1, x_2, \cdots, x_l\}$ is defined as follows:

$$\begin{aligned} y_1 &= 1 \\ y_i &= x_i / x_{i-1}, i = \{2, \cdots, l\} \end{aligned} \tag{6}$$

4.1.2 Training and Testing

The five weeks data in Fig. 3 are divided into the training and testing data sets. The first week data are used as the training set, which is presented as Series 1 in Fig. 3 and the other four weeks data presented as Series 2, 3, 4 and 5 are used for testing. Radial Basic Function (RBF) kernel is chosen as the kernel $K(x, y)$, which can be expressed as:

$$K(x, y) = e^{-\|x-y\|^2/(2\sigma^2)} \tag{7}$$

The parameter σ is chosen to be 2.5, and the trade-off parameter ν in Function (4) is selected to be 0.01, which supposes that a maximum of 1% data points in the training set are abnormal. After training, the model obtained is applied to the testing data set. Fig. 4 and Fig. 5 illustrate the offline monitoring results for week3 and week 4, respectively.

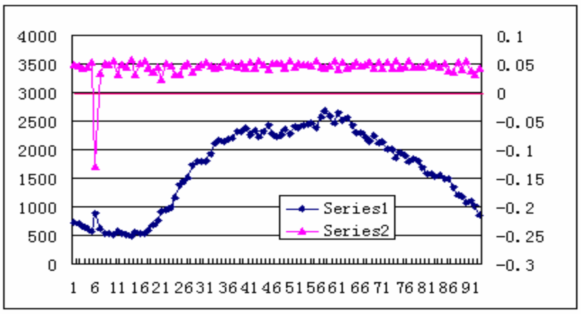

Fig. 4 – Testing results for week 3

In Fig. 4 and Fig. 5, Series 1 is the plot of the original traffic values for one week, and Series 2 presents the results returned by the decision function *f(x)*. These results perfectly match the human visual detection result. It can be seen from the Fig.s that one anomaly at data point 7 is successfully detected in week 3, and three anomalies at data points 7, 9 and 94 are detected in week 4.

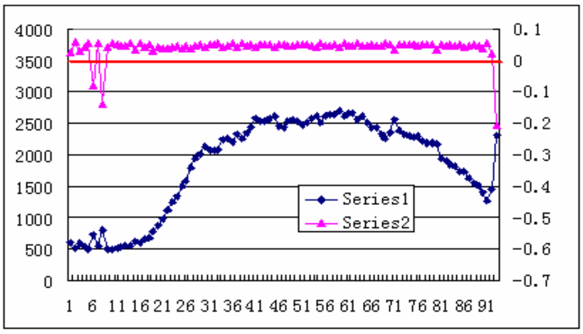

Fig. 5 – Testing results for week4

Some of the returned values from decision function $f(x)=\sum_{i=1}^{N_{sv}}\alpha_i^* K(x_i,x)-\rho$ for week 4 are listed in Table 1. There are altogether three anomalies in week 4 with the returned values being -0.0801, -0.1376 and -0.2056. By comparing the returned value with their original traffic value, it can be concluded that the farther a value derivates from the normal trend, the smaller the returned negative value.

Table 1 – Part of returned values from decision function for offline test on week 4

Part of outputs from function $f(x)=\sum_{i=1}^{N_{sv}}\alpha_i^* K(x_i,x)-\rho$		
Points 1-5	Points 6-10	Points 90-94
0.0250	**-0.0801**	0.0451
0.0594	0.0564	0.0394
0.0284	**-0.1376**	0.0586
0.0424	0.0433	0.0232
0.0547	0.0528	**-0.2056**

4.2 Experiment on Large Dataset

After successfully detecting the anomalies in small data set, the experiment is moving on to a large data set for further testing. The large data set contains the traffic data of 31 continuous days, with 2976 values in total and every 96 values for one day. The original traffic data is shown as the lower plot in Fig. 6.

First gradient feature is extracted for both the training and testing data. The parameter σ is set to be 2.5, and the trade-off parameter ν is 0.01. The results are displayed in the upper part of Fig. 6. 32 abnormal data points are successfully detected using the OCSVM detector.

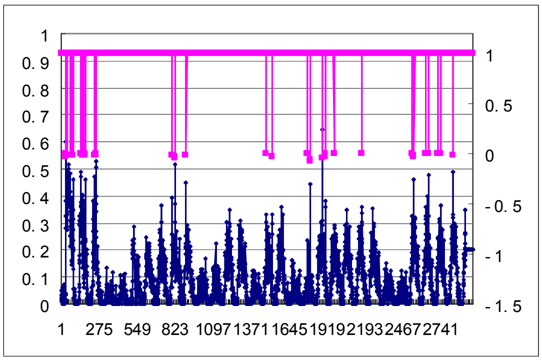

Fig. 6 – Training and testing on a 31 days traffic dataset

4.3 Experiments on Efficiency

As mentioned in introduction, the telecommunication networks are becoming larger and larger, and each operational element will generate performance data in a given time interval. The PM investigation system has to efficient enough to finish all the detection in the given time.

For example, generally, a PM AI module should be capable of processing up to 25-50 million records per hour. This is translated into about 15-20 thousands records per second. But, if the PM AI cannot meet with this enquiry, the system will be in risk.

In order to check the execution time, two larger data sets are selected for detection, with 23,232 and 45,398 samples each. The PC used for training and detection are characterized as: Pentium(R) D CPU 2.8GHz, and 1.99GB of RAM. The training time and detection time are shown in Table 2.

Table 2 – Execution time for detection

	Data set with 23,232 samples	Data set with 45,398 samples
Total execution time (ms)	949	1796
Samples per second	24,480	25,300

The results in table 2 clearly show that the detector can cope with around 25-25 thousands samples per second, the speed exceeds the fundamental enquiries of a PM system. Consider that several CPUs will be adopted for detection, and additionally, multiple threads techniques will be utilized for online detection, the real online PM system can only be even faster.

5 Conclusion

This paper proposes a PM anomaly detection system based on the OCSVM. By solving an optimal problem with slack variables and trade-off parameter, the OCSVM can capture most of the data in a data set as normal in a "small" region, and flag a small part of data as anomalies. The feasibility of OCSVM for telecommunication anomaly detection is verified with the real offline data. Experiments show that the system is accurate and efficient enough to successfully detect the anomalies in the given time interval.

References:

[1] H. Debar, M. Dacier and A. Wespi, "Towards a taxonomy of intrusion detection systems," Computer Networks, 31(8), pp 805-822, April, 1999.

[2] W. Lee, S. Stolfo, and P. K. Chan, "Learning patterns from Unix process execution traces for intrusion detection," in Proc. AAAI Workshop Ai Methods Fraud and Risk Management, Providence, RI, pp.50-56, 1997.

[3] N. Ye, X. Li, Q. Chen, S. Eran, and M. Xu, "Probabilistic techniques for intrusion detection based on computer audit data," IEEE trans. Syst. Man, Cybern., Part A, Syst., Humans, vol. 31, no. 4, pp. 266-174, July, 2001.

[4] Y. Liu, D. Tian and D. Wei, "A wireless intrusion detection method based on neural network," In Proceedings of the 2nd IASTED international conference on Advances in computer science and technology, pp.207-211, Puerto Vallarta, Mexico, January, 2006.

[5] V. Dao and V. Vemuri, "A performance comparison of different back propagation neural networks methods in computer network intrusion detection," Differential of Equations and dynamical Systems, pp. 201-214, 2002.

[6] S. Mukkamala, G. Janoski, and A. Sung, "Intrusion detection using support vector machines," In Proceedings of the high Performance Computing Symposium – HPC 2002, pp. 178-183, San Diego, April, 2002.

[7] W. Hu, Y. Liao, and V.R. Vemuri, "Robust support vector machines for anomaly detection in computer security," in Proc. Int. conf. Machine learning and Applications, Los Angeles, CA, pp. 168-174, June, 2003.

[8] Y. Liao and V. R. Vemuri, "Use of K-nearest neighbour classifier for intrusion detection," Comput. Secur., vol. 21, no. 5, pp. 439-488, 2002.

[9] C. Warrender, S. Forrest, and B. Pearlmutter, "Detecting intrusions using system calls: Alternative data models," in Proc. IEEE Sysmp. Security and privacy, Oakland, CA, pp. 133-145, May, 1999.

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

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

[12] Z. Ji and D. Dusgupta, "Applicability issues of the real-valued negative selection algorithms," In Genetic And Evolutionary Computation Conference, Proceedings of the 8th annual conference on Genetic and evolutionary computation, pp. 111 - 118, 2006.

[13] T. Stibor, P. Mohr, J. Timmis, and C. Eckert, "Is negative selection appropriate for anomaly detection," In Genetic and Evolutionary Computation Conference (GECCO), ACM Press, pp. 321-328, Washington DC, USA, June, 2005.

[14] B. Schölkopf, J. Platt, J. Shawe-Taylor, A.J. Smola, and R. Williamson, "Estimating the support of a high-dimensional distribution," Neural computation, vol. 13, no. 7, pp. 1443-1472, 2001.

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

[16] K. A. heller, K. M. Svore, A. D. Keromytis, and S. J. Stolfo, "One class support vector machines for detecting anomalous windows registry accesses," In 3rd IEEE conference Data Mining Workshop on data Mining for Computer Security, Florida, Nov. 2003.

[17] Q. Tran, H. Duan, X. Li, "One-class Support Vector Machine for Anomaly Network Traffic Detection," The 2nd Network Research Workshop of the 18th APAN, Cairns, Australia, 2004.