Multi Feature Selection based Network Traffic Anomaly Detection Method

Rui Wang¹,a, Jiafu Fang²,b, Zhiye Yang³,c and Haiwei Li⁴,d
¹, ⁴ Department of Information Security, First Research Institute of the Ministry of Public Security of PRC, Beijing China
² Network Security Corps, Liaoning Provincial Public Security Bureau, Shenyang China
³ Network Security Detachment, Beijing Railway Public Security Bureau Tianjin Public Security Bureau, Tianjin China
a ruiwang114@163.com; b13577215@qq.com; c yzy368@163.com; d lhwivan@163.com

Abstract. In this paper, a method is proposed to solve the difficult problem of the training model and the dynamic variability of the deployment environment. Firstly, the network traffic data is converted into numerical value and projected onto histograms of different dimensions to construct detection vectors. Based on the detection vector, some kinds of classifiers are compared. SVDD, which can handle high-dimensional data and has strong generalization ability, is chosen for anomaly detection. Secondly, in order to improve the true positive rate of detection and reduce training time, the classifier is trained continuously and trying various different combinations of features. Finally, a multi-step correlation detection algorithm is adopted to optimize the detection accuracy, and obvious abnormal samples are eliminated from the newly added samples, reducing the training cost and improving the classification accuracy. Through experiments based on a large amount of real network traffic data, the result demonstrate that the proposed method has higher accuracy and lower false alarm rate, and can effectively reduce the training cost.

1. Introduction

With the continuous development of network applications, various network attacks become more serious, which seriously threaten the network security, e.g., virus, worm, distributed denial of service, etc. As an effective network protection method, network traffic anomaly detection can detect unknown attacks, and various types of traffic anomaly detection methods have been proposed by researchers, such as statistical theory, data mining, machine learning and all kinds of hybrid methods [1].

Existing anomaly detection systems [2] [3] are usually constructed based on machine learning methods, and have become a popular field which attract more researchers. As various machine learning [4] theories are widely used, anomaly detection systems have been able to detect more hidden attacks, and classification methods are one of the most important ones. However, these systems have common drawbacks:

• In order to achieve better detection accuracy, a large number of training samples are usually needed, but the workload of collecting and identifying training data set is high.
• In the application of traffic anomaly detection, how to construct a detection vector (classification vector) and choose a classifier has a important influence on the detection effect which often overlooked by researchers.
• Due to the network environment is dynamically changing, and with the advance of time, classification effects of trained classifiers may gradually deteriorate. Retraining can be used to solve this problem, but the cost of retraining classifier is high.
In this paper, a method is proposed to solve the difficult problem of the training model and the dynamic variability of the deployment environment.

Our contributions can be summarized as follows:

- String type IP address of the network traffic data is converted into long type and projected onto a histogram, and using unsupervised classification algorithm [5] for anomaly detection on the network traffic histogram.
- A randomly selecting and combining features method is adopted for continuously training the classification model to improve the accuracy of the anomaly detection system and reduce the training cost.

Based on the problem of the time dependence of the anomaly detection quantity and the serious influence of the abnormal sample on the detection accuracy, we propose a sample selection algorithm optimization scheme.

2. Classifier Selection and Application

2.1. Data Preprocessing

Traffic data histograms [6] are traffic data distributed in a certain dimension approximate situation, in which the traffic data may be the number of network flows, the number of IP packets or the total number of bits, the dimension may be source/destination IP, source/destination port, TCP identification bit, protocol number, length of IP package or flow persistence, timestamp, etc. In this paper, the number of IP packets is recorded from four dimensions of source/destination IP and source/destination port. The principle of classification on the histogram is, in principle, to detect anomalies by measuring differences in the distribution characteristics in order to increase the finer-grained information than entropy.

![Data preprocessing](image)

**Figure 1.** Data preprocessing.

2.2. Classifier

Unsupervised SVM [7] known as SVDD is based on statistical learning theory and can obtain better classifiers with limited training samples. It not only has a strict theoretical basis but also can better solve practical problems such as small samples, nonlinearity, high dimensionality, local minima, etc. SVDD calculates the optimal classifying hyperplane which requires the classifying hyper-plane can separate the two classes without errors and maximize the interval between the two classes, and then ensuring that the classifier's generalization ability is optimized while minimizing the empirical risk. SVDD also creates a hyper-sphere with a small volume in the feature, and contains as many training samples as possible [8].

Both algorithms of SVDD and One Class SVM (OCSVM) are equivalent if all samples lie at the same distance from the origin and are linearly separable from it. Compare with OCSVM:

- SVDD constructs the most suitable sphere that contains one class samples (green square) except for other outliers (red circular) as shown in Figure 2 (a).
- OCSVM separates one class samples (green square) from outliers (red circular) by finding a hyperplane (yellow) of maximal distance from the origin as shown in Figure 2 (b).
SVDD performs outlier detection in a latent feature space by kernelizing Equations as described in paper [5]. For a given kernel \( k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \) with induced Hilbert space \( H \) and feature map \( \phi : \mathcal{X} \to H \), the problem becomes:

\[
\min_{R \in \mathbb{R}_+} \frac{1}{2n} \sum_{i=1}^{n} \xi_i^2 \quad \text{subject to} \quad \sum_{i=1}^{n} \xi_i = R^2 + \frac{1}{n} \sum_{i=1}^{n} \xi_i
\]

Subject to

\[
\| \phi(x_i) - c \|_H^2 \leq R^2 + \xi_i \quad \text{where } i \text{ from } 1 \text{ to } n
\]

This is a convex optimization problem and can be solved efficiently even for several thousand data points. By use of the Representer Theorem [9], one obtains that the sphere’s center point can be expressed as a linear combination \( c = \sum \alpha_i \phi(x_i) \) of the embedded data points. This allows an expression for the decision function depending only on kernel evaluation:

\[
f(x) = R^2 - \| \phi(x) - c \|_H^2
\]

\[
= R^2 - k(x, x) + \sum_{i=1}^{n} \alpha_i \phi(x_i) - \sum_{i,j=1}^{n} \alpha_i \alpha_j k(x, x_i)
\]

Where the last sum does not depend on \( x \) and can be merged with the constant \( R^2 \).
2.3. Feature Selection
Choosing 10 most obvious feature vectors which can distinguish normal categories from abnormal ones, as shown in Table I.

From No. 1 to 10, the test results of different optimal number of features are measured respectively, and the optimal number of features is selected.

| Number | Vector       | Description          |
|--------|--------------|----------------------|
| 1      | src_ip       | source ip            |
| 2      | dst_ip       | destination ip       |
| 3      | src_port     | source port          |
| 4      | dst_port     | destination port     |
| 5      | trans_proto  | transport layer protocol |
| 6      | app_proto    | application layer protocol |
| 7      | upload_pkg   | total upload packages |
| 8      | upload_bytes | total upload bytes   |
| 9      | download_pkg | total download packages |
| 10     | download_bytes | total download bytes |

2.4. Parameter Tuning
The SVDD hyper-parameter $nu$ [10] indicates the upper limit of the classification error rate and the lower limit of the support vector ratio, and the hyper-parameter gamma indicates the influence of a single sample on the hyperplane. We need to define the parameter space and choose the parameters that maximize the AUC score of the training data to establish the optimal model.

2.5. Cross Validation
$K$-fold cross-validation [11] is used for training data under each parameter pair to reduce the unbalanced errors caused by data selection. We tried $K$ from 1 to 10 through real time network traffic data set, and choose $k=5$ which can minimize unbalanced errors.

2.6. Parameter Tuning
True Positive Rate (TPR) score (Value range [0~1], $TPR=TP/ (TP+FN)$) and False Positive Rate (FPR) score (Value range [0~1], $FPR=FP/ (FP+TN)$) to evaluate the precision of the model, the larger the TPR score and the lower the FPR score, the higher the precision, and the better the model.

AUC score [12] (value range [0~1]) is used to as measure of TPR and FPR, which indicates the area under the ROC [13] curve and can comprehensively reflect the quality of the model.

The confusion matrix [14] is used to evaluate the visual test results:
- Confusion matrix[0][0]=TN indicates the number of samples’ real tag is -1 and the prediction tag is also -1.
- Confusion matrix[0][1]=FP indicates the number of samples’ real tag is 1 and the predicting tag is -1;
- Confusion matrix[1][0]=FN indicates the number of samples’ real tag is 1 and the predicted tag is -1;
- Confusion matrix[1][1]=TP indicates the number of samples’ real tag is 1 and the predicting tag is 1.
3. Experiments
In order to verify the validity of the method, experiment is implemented on a real worm network traffic data set. The experimental results demonstrated that the classifier based on SVDD has high accuracy which can effectively detect network abnormal traffic, and can be trained and updated continuously with the arrival of network traffic data.

3.1. Data Set Selection
The data set used in experiment are collected in real network environment. 10000 normal traffic data and 189 abnormal traffic data during a worm virus outbreak were chosen as training and test data set. Each of the data contains the 10 attributes mentioned in Section 2.3.

3.2. Result of Different Combination of Features
As shown in Figure 3, the model achieves the best results when the field app_proto only chosen as the feature. The AUC_score of the training set is 0.9981420785842328, and the AUC_score of the test set is 0.99850099933377745. Though observing the training process of the model, we found that the value of the app_proto field of all virus traffic is 109 or 200, while the value of the app_proto field of only 35 connections in all normal network traffic is 200. Therefore, when the algorithm only chooses the app_proto field as training feature, the model can distinguish normal traffic from abnormal traffic, and the AUC_score of the model reaches the maximum value.

When all 10 fields are chosen as features, the result of the test set is decreased. The result indicates that the model has overfitting, and demonstrates that the number of selected features is not as much as possible.

![Figure 3. AUC_score of different combination of features](image)

3.3. Parameter Tuning
We tuned parameters through try different combination of nu and gamma to optimize the SVDD model. The range of parameters is shown in Table II. The debugging process lasted about 69 seconds.

When the parameter best_feature close to 1, nu equals 0.25 and gamma equals 0.01, the effect of the model reaches best.

| Parameter | Range         |
|-----------|---------------|
| nu        | [0.1, 0.25, 0.5, 0.75, 1] |
| gamma     | [1, 0.1, 0.01] |
3.4. Experiments Results
The experiment results are shown in Table II, and the confusion matrix on the test set is shown in Figure 4. The AUC_score and TPR_score on training set reaches 0.998142 and 0.996284 respectively. The number of true positive samples is 2993, number of true_negative samples is 9, number of false_negative is 55, and no false_positive samples. The TRP_score on test data set reach 0.996248; demonstrate that the proposed method has higher accuracy and lower false alarm rate with high training efficiency.

| Data Set | AUC_score | TPR_score | FPR_score |
|----------|-----------|-----------|-----------|
| Training | 0.998142  | 0.996284  | 0         |
| Test     | 0.998500999 | 0.997001998 | 0         |

Table 4. Confusion matrix on the test set

4. Conclusion
The Experiments results demonstrate that the proposed method has higher accuracy and lower false alarm rate with high training efficiency. Although the model reaches the best effect when only choose app_proto as feature, it is found that when tuning parameters, the model effect is not stable under different parameter, and this situation demonstrates that the model generalization ability of selecting only one feature is not strong.

When choosing three feature fields (src_port, app_proto, dst_port) or four feature fields (src_ip, src_port, app_proto, dst_port), the model is stable with high accuracy, but the efficiency declined significantly. Therefore, when training model on a larger data set, parameter tuning should choose an appropriate number of features to achieve a trade-off between accuracy and efficiency.

5. References
[1] Patcha A, Park J M. An overview of anomaly detection techniques: Existing solutions and latest technological trends [J]. Computer networks, 2007, 51(12): 3448-3470.
[2] Depren O, Topallar M, Anarim E, et al. An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks [J]. Expert systems with Applications, 2005, 29(4): 713-722.
[3] Zheng L M, Zou P, Jia Y, et al. How to extract and train the classifier in traffic anomaly detection system [J]. Jisuanji Xuebao(Chinese Journal of Computers), 2012, 35(4): 719-729.
[4] Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine learning in Python [J]. Journal of machine learning research, 2011, 12(Oct): 2825-2830.

[5] Tax D M J, Duin R P W. Support vector domain description [J]. Pattern recognition letters, 1999, 20(11-13): 1191-1199.

[6] Claffy K C, Polyzos G C, Braun H W. Application of sampling methodologies to network traffic characterization[C]//ACM SIGCOMM Computer Communication Review. ACM, 1993, 23(4): 194-203.

[7] Hearst M A, Dumais S T, Osuna E, et al. Support vector machines [J]. IEEE Intelligent Systems and their applications, 1998, 13(4): 18-28.

[8] Banerjee A, Burlina P, Meth R. Fast hyperspectral anomaly detection via SVDD[C]//Image Processing, 2007. ICIP 2007. IEEE International Conference on. IEEE, 2007, 4: IV-101-IV-104.

[9] Schölkopf B, Herbrich R, Smola A J. A generalized representer theorem[C]//International conference on computational learning theory. Springer, Berlin, Heidelberg, 2001: 416-426.

[10] Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine learning in Python [J]. Journal of machine learning research, 2011, 12(Oct): 2825-2830.

[11] Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection[C]//IJCAI. 1995, 14(2): 1137-1145.

[12] Lobo J M, Jiménez - Valverde A, Real R. AUC: a misleading measure of the performance of predictive distribution models [J]. Global ecology and Biogeography, 2008, 17(2): 145-151.

[13] Davis J, Goadrich M. The relationship between Precision-Recall and ROC curves[C]//Proceedings of the 23rd international conference on Machine learning. ACM, 2006: 233-240.

[14] Townsend J T. Theoretical analysis of an alphabetic confusion matrix [J]. Perception & Psychophysics, 1971, 9(1): 40-50.