Application of SVM in anomaly detection based on sampling and feature extraction

Kaikun Dong, Jiao Shi, Li Guo and Fang Yuan
School of Computer Science and Technology, Harbin Institute of Technology at Weihai, Weihai, China

Corresponding author and e-mail: Li Guo, kaikun.dong@gmail.com

Abstract. In order to identify the four types of attacks in the KDD99 data set, an anomaly detection model based on supporting vector machines is proposed. The model uses supporting vector machines to train five classifiers and detects the attacks by integrating the classification results of the five classifiers. Aiming at the problem that the detection model has a low recall rate when detecting some categories, a feature extraction method based on abnormal proportion analysis is proposed to extract effective feature combinations for each classifier, and a sampling technique is used to preprocess the training set of each classifier. Experimental results show that the improved anomaly detection model can not only improve the performance of each classifier, but also improve the comprehensive discrimination performance of the five classifiers.

1. Introduction
As an active defense technology, intrusion detection technology can effectively detect the violation of security policies in computer systems. From a technical perspective, intrusion detection is divided into misuse detection and anomaly detection. Compared with the misuse detection, anomaly detection can detect unknown types of attacks and is currently a research hotspot in the field of intrusion detection. Among the anomaly detection technologies, the detection technology based on machine learning is commonly used. The methods used mainly include clustering algorithms and classification algorithms.

In this paper, the supporting vector machine (SVM) is used to train the anomaly detection model. Because the training data set plays a key role in the final detection effect, this paper uses the authoritative public data set KDD99 data set in the field of intrusion detection to train the model. Based on the KDD99 data set and machine learning algorithms, there are many research achievements in anomaly detection. Among them, the recall rate of the r2l and u2r categories is generally low, which is a common concern of many literatures. Based on the K-means clustering algorithm, Jia Fan et al. selected different feature combinations for different attack types to detect anomalies [1]. Sabhnani et al. comprehensively used neural network, K-means and Gaussian process classification algorithms to train a classifier for each attack type [2], which achieved a good comprehensive detection effect, but in the r2l and u2r categories, the recall rate is low, at 29.8% and 9.6%, respectively. Wang et al. combined an association rule and rough set to design an anomaly detection model [3], but the recall rates on the r2l and u2r categories are only 7.9% and 3.8%, respectively. Toosi et al. combined fuzzy neural network and genetic algorithm to identify four kinds of attacks in the KDD99 dataset [4]. The recall rates in the r2l and u2r categories have been improved, reaching 31.5% and 14.1%, respectively.
2. Anomaly detection based on multi-class SVM

2.1. The multi-classification method based on SVM

Classical SVM can only deal with binary classification problems. To process multi-class classification problems, the SVM needs to be further processed. Common multi-classification methods of combined classifiers include one-to-one SVMs (1-v-1 SVMs), one-to-many SVMs (1-v-r SVMs), and decision-oriented acyclic graph SVMs (DAG SVMs) [5], error correction coding SVMs (ECC SVMs) [6], etc.

Assuming that the N classification problem needs to be solved, a one-to-many SVM trains a total of N classifiers, and each classifier trains a certain class as a positive class, and the sum of all other classes as a negative class. When testing a sample, pass it through each classifier, if only one classifier judge it positive, then the final classification result is the positive category corresponding to the classifier; if multiple classifiers judge it positive, then the final result is comprehensively determined according to the confidence of each classifier that judges it positive; if no classifier judge it positive, take the positive class corresponding to the classifier with the smallest confidence as the final classification result.

2.2. Anomaly detection model

In this paper, the one-to-many SVM has been used to process multi-classification problems in anomaly detection. The specific idea is: training a classifier for each category in the KDD99 data set, and record them as normal classifier, dos classifier, probe classifier, r2l classifier, u2r classifier, and each classifier takes the sample to be divided It is a positive sample, and the remaining samples are negative samples. The final classification result is comprehensively determined according to the confidence of each classifier. Based on the idea, an anomaly detection model is proposed as shown in Figure 1.

3. Improved anomaly detection based on sampling technology and feature extraction technology

3.1. Feature extraction method based on abnormal proportion analysis

In this paper, the training set and test set of the KDD99 data set are merged, and the combined data set is quantified and standardized. The element in row i and column j of the merged and preprocessed data set is $X_{ij}$, then when the absolute value of $X_{ij}$ is greater than a certain threshold, the element is an outlier. The threshold used in this article is 3, that is, when $|X_{ij}| > 3$, $X_{ij}$ is an outlier. Assuming the number of outliers in the jth column is N, then the proportion of a certain category among the N outliers is the abnormal proportion of the category in the jth column. For example, the abnormal proportion of r2l attack types in 41 dimensions is shown in Figure 2.
Figure 2. R2L attack in 41 dimensions of abnormal ratio bar graph.

Since \(2^{41}-1\) feature combinations can be derived from 41 features, it is very time-consuming to traverse each combination to find the best feature combination, so a feature extraction method is proposed based on abnormal proportion analysis to quickly find a better feature combination.

Step1: Remember that the feature combination with r2l category anomaly rate greater than 0% is \(G_{r2l}\), the combination of features other than \(G_{r2l}\) is \(G_{other}\), and \(G_{start} = G_{r2l}\).

Step2: Delete a feature \(s\) in \(G_{start}\) and measure the accuracy rate and recall rate of the positive class under \(G_{start}\). If the accuracy rate does not decrease by more than 0.05, the recall rate is improved, or if the recall rate is unchanged, the accuracy rate is increased, \(s\) is not restored, otherwise restore \(s\) to \(G_{start}\). Continue to delete another feature that has not been deleted in \(G_{start}\) and repeat the above process until all features in \(G_{start}\) have been tested;

Step3: Add a feature \(o\) in \(G_{other}\) to \(G_{start}\), and measure the accuracy and recall rate of the positive class under \(G_{start}\). If the recall rate is increased when the accuracy rate does not decrease by more than 0.05, or if the recall rate is unchanged, the accuracy rate is improved, \(o\) is not deleted, otherwise \(o\) is deleted from \(G_{start}\). Continue to add a feature that has not been added in \(G_{other}\) and repeat the above process until all features in \(G_{other}\) have been tested.

Step4: Use \(G_{start}\) as a feature combination of r2l classifier.

The effective feature combination of each classifier extracted based on abnormal proportion analysis is shown in Table 1, the first feature is marked as 0, and so on.

| Classifier  | Extracted feature combination                      |
|-------------|---------------------------------------------------|
| normal      | 0,1,2,3,4,5,6,7,8,9,10,11,13,14,15,17,18,22,23,24,25,26,27,29,30,31,32,33,35,36,37,38,39,40 |
| dos         | 2,3,4,5,6,8,9,10,11,12,13,14,15,16,17,18,19,21,22,23,28,29,30,31,32,33,34,35,36          |
| probe       | 0,2,3,4,6,10,13,22,23,26,27,29,30,36,39,40                                             |
| r2l         | 1,2,5,6,9,10,13,14,16,17,18,19,21,22,30,31,32,35,37                                      |
| u2r         | 0,2,4,9,13,14,16,20,21,23,30,31                                                            |

Table 1. Feature combinations after feature extraction based on abnormal proportion analysis.

3.2. Optimization of classifier based on sampling technique

3.2.1. Sampling technology. The sampling algorithms used in this article are as follows.

a. OSS algorithm

The specific implementation steps of the OSS algorithm are as follows:
Step 1: Remember that the current sample set is S, and the majority sample set is C, and the remaining sample set after the removal of C is D;

Step 2: Randomly select a sample from C and put it in D.

Step 3: Train a kNN classifier (usually 1NN classifier) with the samples in D.

Step 4: Traverse the samples in S to classify with the trained kNN classifier, and put the misclassified samples into D. The kNN classifier changes with the addition of samples in each iteration.

Step 5: Delete the samples belonging to the majority class in the Tomek link pair in D.

Step 6: Take D as the final sample set after undersampling.

b. ENN algorithm

The specific execution steps of the RENN algorithm are as follows:

Step 1: Record the original sample set as S, the undersampled sample set as D, and initialize D as S;

Step 2: Iterate through D and find its k-nearest neighbor for each sample. If the sample has the same type as all samples in the k-nearest neighbor, then keep the sample, otherwise delete the sample from the training set.

Step 3: Repeat Step 1 and Step 2 until no samples in the training set are deleted.

c. SMOTE algorithm

The specific implementation steps of the SMOTE algorithm are as follows:

Step 1: Remember that the number of minority samples is $N$, one of the minority samples is $x_i$, $i=1,2,...,N$, and $x_i$ is an $n$-dimensional vector in $R^n$;

Step 1: Calculate the number of new samples $M$ required for this oversampling according to the target number of oversampling, and calculate based on each $x_i$, the average required new sample size is $D$.

Step 2: For each $x_i$ use a formula (1) to generate a new sample $y_i$, execute $D$ times in total, and add the generated new sample to the sample set. Where $xl_i$ is a sample that randomly selected from the same kind of k-nearest neighbors of $x_i$, and $rand(0,1)$ is a random number between 0 and 1.

$$y_i = x_i + (xl_i - x_i) \times rand(0,1)$$

(1)

d. ADASYN algorithm

The specific implementation steps of the ADASYN algorithm are as follows:

Step 1: Remember that the number of minority samples is $N$, one of the minority samples is $x_i$, $i=1,2,...,N$, and $x_i$ is an $n$-dimensional vector in $R^n$;

Step 2: Calculate the number $M$ of new samples required for this oversampling according to the target number of oversampling;

Step 3: The number of samples of the majority class in k nearest neighbors of $x_i$ is $\alpha_i$, the proportion of the majority class of k nearest neighbors of $x_i$ is $r_i, i=1,2,...,N$, the ratio after normalizing $r_{i} = \hat{r}_{i}$, $\hat{r}_{i}$ is the ratio of the number of new samples that need to be generated based on each $x_i$ to the total number of new samples that need to be generated, and the number of new samples generated based on $x_i$ is $d_i$. The calculation of $r_i, \hat{r}_{i}, d_i$ is shown in formula (2);

$$r_i = \frac{\alpha_i}{k}, \quad \hat{r}_{i} = \frac{r_i}{\sum r_i}, \quad d_i = \hat{r}_{i} \times M$$

(2)

Step 4: For each $x_i$ use a formula (4) to generate a new sample $y_i$, executed $d_i$ times, add the generated new sample to the sample set. Where $xl_i$ is a sample that randomly selected from the same kind of k-nearest neighbors of $x_i$, and $rand(0,1)$ is a random number between 0 and 1.
3.2.2. **Classifier optimization method.** Since 10% of the training and the test set of KDD99 contain a lot of repeated data, model training will cause the algorithm to prefer those frequently appearing samples, making the knowledge learned from a few samples less. The performance of the test model based on repeated test sets will also make the final test results lack of objectivity. Therefore, this paper first performs a deduplication operation on both the training set and the test set. Then the numerical and standardization operations of the two data set are performed. In addition, the OSS algorithm is used to remove the redundant samples and noise samples in the normal and dos type data in the training set.

4. **Experimental results**

The positive precision (+), positive recall (+), positive f1-score (+), g-means, and auc of the five major classifiers are trained independently and tested with the test set. The size before feature extraction and sampling is shown in Table 2, and the size after feature extraction and sampling is shown in Table 3, where the performance indicators before feature extraction and sampling are obtained from the data set after the OSS algorithm has been used to undersample the normal and dos data.

**Table 2.** Indexes of each independent classifier before feature extraction and sampling.

| Classifier | precision(+) | recall(+) | f1-score(+) | g-means | auc     |
|------------|--------------|-----------|-------------|---------|---------|
| normal Classifier | 0.918 | 0.995 | 0.955 | 0.954 | 0.958 |
| dos Classifier | 0.994 | 0.661 | 0.794 | 0.930 | 0.751 |
| probe Classifier | 0.216 | 0.650 | 0.324 | 0.461 | 0.931 |
| r2l Classifier | 0.917 | 0.113 | 0.202 | 0.941 | 0.703 |
| u2r Classifier | 0.600 | 0.056 | 0.102 | 0.774 | 0.768 |

**Table 3.** Indexes of each independent classifier after feature extraction and sampling.

| Classifier | precision(+) | recall(+) | f1-score(+) | g-means | auc     |
|------------|--------------|-----------|-------------|---------|---------|
| normal Classifier | 0.936 | 0.994 | 0.964 | 0.962 | 0.959 |
| dos Classifier | 0.985 | 0.939 | 0.962 | 0.980 | 0.983 |
| probe Classifier | 0.916 | 0.687 | 0.785 | 0.952 | 0.899 |
| r2l Classifier | 0.916 | 0.449 | 0.603 | 0.947 | 0.835 |
| u2r Classifier | 0.838 | 0.144 | 0.246 | 0.914 | 0.837 |

The five major classifiers are integrated according to the principle of one-to-many SVM and tested with the deduplicated test set. The comparison of precision, recall and f1-score of each category before and after feature extraction and sampling is shown in Table 4.

**Table 4.** Performance comparison of the integrated multi-classifier before and after feature extraction and sampling.

| category | Before feature extraction and sampling | After feature extraction and sampling |
|----------|---------------------------------------|---------------------------------------|
|          | precision | recall | f1-score | precision | recall | f1-score |
| normal   | 0.911     | 0.995  | 0.951    | 0.937     | 0.992  | 0.964    |
| dos      | 0.994     | 0.663  | 0.795    | 0.989     | 0.956  | 0.972    |
| probe    | 0.213     | 0.695  | 0.327    | 0.823     | 0.702  | 0.758    |
| r2l      | 0.943     | 0.165  | 0.281    | 0.882     | 0.450  | 0.596    |
| u2r      | 0.583     | 0.065  | 0.117    | 0.769     | 0.140  | 0.236    |

It can be seen from Table 4 that after the multi-classifier integrated according to the principle of one-to-many support vector machine, which undergoes feature extraction, sampling of the training set and optimization of the parameters of the classifier, none of the decline indicators exceeded 6%, except for the recall rate of the normal category decreased by 0.3%, the precision rate of the dos class decreased by 0.5%, and the precision rate of the r2l class decreased by 6.1%. The remaining performance indicators were improved and unchanged, and all F1 values were improved. Among them, the recall rate of the dos class has increased by 29.3%, the precision rate of the probe class has increased by 61.0%, the recall rate of r2l has been increased by 28.5%, and the precision rate of u2r has been increased by 18.6%, all of which have been improved by more than 18%.
The recall rates of r2l classifier and u2r classifier in Table 3 have reached 45.0% and 14.0% respectively. Although the recall rate of the r2l attack and the u2r attack has improved, it is still low for two reasons: one is that the sample size of the two types of attacks in the training set is too small, and only accounts for 2.7% of the deduplication training set and 0.1% of the training set; the second is that 2l attacks and u2r attacks do not have frequent sequence patterns in data records like dos attacks, but are generally embedded in the data load of data packets and have few information about the formatted network connection, a single data packet is almost the same as a normal connection.

5. Conclusions
To realize the classification of the five types of data in KDD99 based on SVM, this paper uses a one-to-many SVM multi-classification method to train five classifiers, and comprehensively determines the type of a sample by the output and confidence of the five classifiers. The anomaly detection model composed of these five classifiers has a low recall rate. In order to improve the performance of the model, this paper extracts an effective feature combination for each classifier based on anomaly ratio analysis, and balances the training set of each classifier except the u2r classifier based on the sampling technique. Experimental results show that the feature extraction method based on abnormal proportion analysis and the classifier optimization method based on sampling technique effectively improve the performance of each classifier, and improve the performance of the integrated five classifiers for multiple classifiers.

References
[1] JIA Fan, YAN Yan, ZHANG Jiaqi. K-means based feature reduction for network anomaly detection[J]. J Tsinghua Univ (Sci & Technol), 2018, 58(2), pp 137–142.
[2] Sabhnani M R, Serpen G. Application of machine learning algorithms to KDD intrusion detection dataset within misusedetection context [EB/OL]. http://neuro.bstu.by/ai/Todom/My_research/Papers-0/For-research/D-mining/Anomaly-D/KDD-cup-99/mlmto3.pdf.
[3] Wang Xuren, He Famei, Xu Rongsheng. Modeling intrusion detection system by discovering association rule in rough set theory framework[C]. Proceedings of the International Conference on Computational Intelligence for Modeling Control and Automation. Washington, DC: IEEE Computer Society, 2006 p 24.
[4] Toosi AN, Kahani M. A new approach to intrusion detection based on an evolutionary soft computing model using neuron-fuzzy classifiers [J]. Computer Communications, 2007, 30, pp 2201–2212.
[5] Platt J C, Cristianini N, Shawe-Taylor J. Large Margin DAGs for Multiclass Classification. In Advances in Neural Information Processing Systems, MIT Press, 2000, (12), pp 547–553.
[6] Dietterich T G, Bakiri G. Solving Multiclass Learning Problem via Error-Correcting Output Codes [J]. Journal of Artificial Intelligence Research, 1995, (2), pp 263–286.