Application of machine learning in BGP anomaly detection

IOP Conf. Series: Journal of Physics: Conf. Series **1176** (2019) 032015 doi:10.1088/1742-6596/1176/3/032015

Application of machine learning in BGP anomaly detection

Xianbo Dai[1,2,*], Na Wang[1,2] and Wenjuan Wang[1,2]

[1]Zhengzhou Information Science and Technology Institute, Zhengzhou, China
[2]Henan Key Laboratory of Information Security, Zhengzhou, China

*Corresponding author e-mail: beyond_dxb@126.com

Abstract. Aiming at the problem of BGP anomalies, a support vector machine-based BGP anomaly detection method (SVM-BGPAD) is proposed. We first employ feature selection algorithm based on Fisher linear analysis and Markov random field technology to select features that can maximize the distance among classes and minimize the distance within the class, and then use grid search and cross-validation methods to optimize the parameters of SVM model. We evaluate the performance of classification model with linear, polynomial, RBF and Sigmoid kernels. The results are compared based on accuracy and F1-Score. Experimental results show that the model based on RBF kernel function can achieve the best classification accuracy of 91.36% and F1-Score of 96.03%.

1. Introduction

The Border Gateway Protocol (BGP) [1] plays an important role in establishing and maintaining the network accessibility among all Autonomous Systems (AS), and plays a vital role in the reliable and stable operation of the entire Internet. However, prefix hijacks, misconfigurations, link failures, and even worm attacks can easily affect the performance of the global Internet BGP, which may destroy the accessibility and stability of the global Internet [2-3]. In recent years, BGP anomalies have occurred frequently. For example, in August 2012, Canadian operator Dery Telecom (AS46618) leaked all its routing tables to its provider Bell Canada (AS577) due to a misconfiguration, resulting in a “BGP updates storm” that disrupted the stability and accessibility of partial networks [4]. In April 2017, the Swiss operator SwissIX (AS 200759) initiated a prefix hijacking attack for the same reason, affecting the accessibility of 576 AS and 3431 prefixes worldwide, including Google, Amazon, Twitter, Apple, etc. [5]. If the occurrence of BGP anomalies can be detected by taking effective countermeasures, the impact of BGP anomalies on global Internet accessibility and stability will be significantly reduced.

According to the consequences of the anomaly events, BGP anomalies can be divided into data flow hijacking anomalies and BGP update messages explosion anomalies. Data flow hijacking anomalies can lead to the redirection of the victim network data flow or/and a traffic black hole, which destroys the accessibility of the victim network. BGP update messages explosion anomalies will result in a large number of BGP update messages generated in a very short time, which destabilizes the stability of the global Internet. In this paper, we mainly focus on the detection of BGP update messages explosion anomalies. At present, BGP anomaly detection methods are classified into five categories, namely statistical pattern recognition, validation of BGP updates based on historical BGP data, reachability check, time series analysis, and machine learning [3]. Statistical pattern recognition method [6] uses statistical probability theory for pattern recognition, and determines the anomalies according to the distance function between modes. It can detect data flow hijacking anomalies and update messages explosion anomalies simultaneously. However, this method faces the difficulty of

correctly estimating the distribution of high-dimensional data, with a low detection speed. In practical applications, the threshold of model parameters needs to be manually determined. Validation of BGP updates based on historical BGP data method [7] and reachability check method [8] can only detect data flow hijacking anomalies. The former uses historical data to detect BGP abnormal routes, while the latter verifies the reachability based on the reachability verification result of the target prefixes. Time series analysis method and machine learning method are capable of detecting update messages explosion anomalies. Time series analysis method [9] treats the BGP update messages as multidimensional time series and implements anomaly detection by selecting an appropriate sliding time window. However, it is difficult for the method to determine the size of the time window. If the time window is too small, the amount of information available to the model will be insufficient. Otherwise, the model will be insensitive to local anomalies, and the false negative rate will increase. In recent years, machine learning methods have been applied to the field of BGP anomaly detection. From the perspective of machine learning, the BGP anomaly detection problem can be abstracted into a two-class problem [10]. For example, Al-Rousan et al. [11] extracted 37 features from the public routing dataset, and achieved a classification accuracy of 81.5% by constructing a hidden Markov model. Li et al. [12] employed decision tree and fuzzy rough set method to select features, and then constructed the extreme learning machine classifier model based on feature subsets, achieving a classification accuracy of 80.08%.

Aiming at the problem of BGP update messages explosion anomalies, a support vector machine-based BGP anomaly detection method is proposed. This method detects BGP update messages explosion anomalies by constructing a SVM classification model. Firstly, the BGP raw data is transformed into a set of features by feature extraction process. After data standardization, Fisher-Markov Selector feature selection algorithm is used to select a feature subset to reduce the influence of noise and redundant data. Considering the imbalance of the training data set, the status of the two classes of samples is balanced by giving greater weight to the smaller one. In particular, the parameters of the model are optimized by grid search and cross-validation methods to improve the performance of the classification model. The SVM classification models with different kernel functions are constructed to find the optimal kernel function suitable for the BGP update messages augmented anomaly detection problem. In addition, we evaluate the performance of the model with F1-Score and accuracy to enhance the validity and scientificity of conclusion. In this paper, we conducted detailed experiments from datasets combination, feature selection and kernel function. The experimental results show that the $SVM_5 - D$ model based on Fisher-Markov Selector feature selection and RBF kernel function achieves the best classification accuracy of 91.36% and F1-Score of 96.03%, and the classification accuracy is approximately 10% higher than the existing machine learning-based anomaly detection method [11-14]. It can be seen that the SVM-BGPAD method has a good classification performance and can be used for BGP anomaly detection.

2. Feature engineering

Data and features affect the classification performance of the model and determine the upper limit of machine learning. So we use feature extraction to transform the BGP raw data into a set of features with obvious physical or statistical significance. The 37 features obtained through the feature extraction process [11] are shown in Table 1, and their values are calculated in one minute intervals.

Table 1. Feature extraction.

Feature	Definition	Category
1	Number of announcements	*volume*
2	Number of withdrawals	*volume*
3	Number of announced NLRI prefixes	*volume*
4	Number of withdrawn NLRI prefixes	*volume*
5	Average AS-PATH length	*AS-path*
6	Maximum AS-PATH length	*AS-path*
7	Average unique AS-PATH length	*AS-path*
8	Number of duplicate announcements	*volume*
9	Number of duplicate withdrawals	*volume*
10	Number of implicit withdrawals	*volume*
11	Average edit distance	*AS-path*
12	Maximum edit distance	*AS-path*
13	Inter-arrival time	*Volume*
14-24	Maximum edit distance = n, where $n = (7, \ldots, 17)$	*AS-path*
25-33	Maximum AS-path length = n, where $n = (7,\ldots,16)$	*AS-path*
34	Number of IGP packets	*volume*
35	Number of EGP packets	*volume*
36	Number of incomplete packets	*volume*
37	Packet size (B)	*volume*

However, the SVM classification model constructed based on high-dimensional features increases the computational complexity, and the noise data also reduces the classification accuracy of the model. Therefore, we further searches for the feature subsets with the best classification ability from 37 features. Feature selection improves the classification performance of the model while reducing computational complexity by eliminating redundant or uncorrelated features. The Fisher-Markov Selector feature selection algorithm [15] can select features that can maximize the distance among classes and minimize the distance within the class based on Fisher linear analysis and Markov random field technology. The feature selection process and the classification process are independent of each other. The correlation is measured only according to the intrinsic properties of data, and the features of each dimension are weighted by the ability to distinguish categories. The method not only ensures global optimization, but also has low computational complexity suitable for processing large-scale data. Therefore, we employ the Fisher-Markov Selector algorithm to select the subset from 37 features obtained through the feature extraction process. Algorithm pseudo code is shown in algorithm 1.

Algorithm 1 Fisher-Markov Selector Feature Selection Algorithm

Input： feature matrix $X=[x_1,...,x_n]\in R^{p\times n}$ ， class label $Y=[y_1,...,y_n]$ ， $y_k\in\{w_1,...,w_g\}$, $(k=1,...n)$, parameter γ,β

Output：feature selector $\alpha^*=[\alpha_1^*,...,\alpha_p^*]^T\in\{0,1\}^p$

1: **begin**

2: **for** $j=1\to p$ **do**

3: $$\theta_j=\frac{1}{n}\sum_{i=1}^{g}\frac{1}{n_i}\sum_{u,v=1}^{n_i}x_{uj}^{(i)}x_{vj}^{(i)}-\frac{\gamma}{n}\sum_{i=1}^{n}x_{ij}^2+\frac{\gamma-1}{n^2}\sum_{u,v=1}^{n}x_{uj}x_{vj}$$

4: **if** $\theta_j>\beta$ **then**

5: $\alpha_j^*=1$

6: **else**

7: $\alpha_j^*=0$

8: **end if**

9: **return** α_j^*, θ_j

10: **end for**

11: **end**

3. SVM Classification Model

3.1. Kernel Function

In 1995, Cortes and Vapnik formally proposed SVM [16], which is a machine learning algorithm with a solid theoretical foundation. SVM has largely overcome the problems of "dimensional disaster" and "over-learning", and shown many unique advantages in solving nonlinear and high-dimensional pattern recognition problems, thus quickly became one of the popular technologies of machine learning.

For linear indivisible problem, the basic idea of SVM is to map the sample space to a high-dimensional feature space by using the kernel function satisfying the Mercer condition, and find a division hyperplane in the high-dimensional feature space, so that the sample can be linearly separable. The Mercer condition converts the corresponding optimization problem into a convex problem, so there is no local minimum. SVM classification models can be constructed using different kernel functions. Table 2 lists several commonly used kernel functions. Pattern recognition theory has proved that if the original space is a finite dimension, that is, the number of attributes is limited, there must be a high-dimensional feature space to make the sample separable. In this paper, we will evaluate the performance of SVM models based on linear, polynomial, RBF and Sigmoid kernels in the experiment to find the optimal kernel function suitable for the BGP update messages explosion anomaly detection problem.

Table 2. Kernel function.

Kernel Function	Expression
linear kernel	$k(x_i, x_j) = x_i^T x_j$
polynomial kernel	$k(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$
RBF kernel	$k(x_i, x_j) = \exp(-\gamma \left\| x_i - x_j \right\|^2), \quad \gamma > 0$
sigmoid kernel	$k(x_i, x_j) = \tanh(\gamma \mathrm{x}_i^T \mathrm{x}_j + r)$

3.2. Grid Search and Cross Validation

The performance of the SVM model depends on two important parameters, *C* and *gamma*. *C* is called a penalty factor, indicating tolerance to error. The appropriate value of the parameter *C* is of great significance to the improvement of the classification accuracy and generalization ability of the model. And *gamma* is a parameter in the polynomial, RBF, and Sigmoid kernels, which implicitly determines the distribution of data after mapping to the new feature space.

Due to the imbalance of the datasets, the traditional classification algorithm with the overall classification accuracy will pay too much attention to the majority class, resulting in a decrease in the classification performance of the minority class. Therefore, we assign weights to positive and negative samples according to the ratio of the samples to effectively solve the imbalance problem.

In addition, we use grid search and cross-validation to optimize parameters. The search range of *C* and *gamma* is divided into grids according to values, and each point in the grid represents a parameter combination scheme. At each point, the total training set is divided into N subsets, of which N-1 subsets are used as training sets, and the remaining one is used as a test set. When N subsets have been tested, the N-fold cross-validation is used to verify the classification accuracy. After traversing all the points in the grid, the point with the highest average value of the classification accuracy is the corresponding optimal parameter combination. It should be pointed out that since both *C* and *gamma* select a limited and discrete search range, the optimal combination of parameters is likely to be only a local optimal solution.

4. Experiment and Analysis

4.1. Evaluation Index

As mentioned above, due to the imbalance of the datasets, it is not appropriate to measure the classification effect of the model only by the classification accuracy. Therefore, we use the classification accuracy and F1-Score to comprehensively evaluate the performance of the model.

4.1.1. Accuracy. The classification accuracy is calculated based on the classification confusion matrix (as shown in Table 3). For binary classification problem, it can be divided into four cases: *TP*, *FN*, *FP*, and *TN*. *TP* indicates that the sample is positive and the prediction result is also positive; *FN* indicates that the sample is positive, but the false prediction is negative; *FP* indicates that the sample is negative, but the prediction result is positive; *TN* indicates that the sample is negative, and the prediction result is negative. Obviously, the classification accuracy can be defined as:

$$Accuracy = \frac{TP + TN}{TP + FN + FT + TN} \tag{1}$$

Table 3. Confusion matrix.

Actual Class	Predicted Class	
	Anomaly (positive)	Regular (negative)
Anomaly (positive)	*TP*	*FN*
Regular (negative)	*FP*	*TN*

4.1.2. F1-Score. F-Score is the weighted average harmonic value of precision and recall. The precision is for our predictions, which indicates how many of the samples with positive predictions are true positive samples. The recall is for our original sample, which indicates how many positive examples in the sample are predicted correctly. Precision and recall are mutually constrained in most cases. Precision, recall and F-Score are defined as follows:

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP+FP} \\ \text{Recall} &= \frac{TP}{TP+FN} \\ \text{F-Score} &= (1+\beta^2)\frac{\text{Precison}\times\text{Recall}}{\beta^2\,\text{Precison}+\text{Recall}} \end{aligned} \tag{2}$$

where $\beta > 0$ measures the importance of precision and recall. In this paper, let $\beta = 1$, which means that precision and recall are equally valued. Correspondingly, F-Score is also called F1-Score, and its calculation formula is:

$$\text{F1-Sore} = 2\times\frac{\text{Precison}\times\text{Recall}}{\text{Precison}+\text{Recall}} \tag{3}$$

4.2. Data Preprocessing

4.2.1. Experimental Datasets. The BGP raw data during worm attacks such as Slammer [17], Nimda [18] and Code Red I [19] are collected from AS513 (RIPE RIS, rcc04, CIXP, Geneva). At the same time, we downloaded the BGP regular datasets from RIPE NCC [20] and the BCNET Network Operations Center [21] in Vancouver, Canada. The libBGPdump tool [22] is used to convert MRT [23] to ASCII format. We parse the ASCII file based on the tools written in C# and extract 37 features sampled every minute in 5 days, producing 7,200 samples for each anomaly event.

4.2.2. Standardization. In order to eliminate the effects of dimension and numerical size, data standardization is required so that different features can be compared and weighted. We use the Z-Score data standardization method, as shown in equation (4), where μ represents the population mean and σ represents the population standard deviation.

$$x' = \frac{x-\mu}{\sigma} \tag{4}$$

In this paper, since the BGP datasets come from sampling statistics, we replace the population mean and population standard deviation with sample mean and sample standard deviation, respectively. The standardization method is as shown in equation (5), where $\bar{x}$ represents the sample mean and *S* represents the sample standard deviation.

$$x' = \frac{x-\bar{x}}{S} \tag{5}$$

4.3. Results and Analysis

4.3.1. Experiment Environment. We conducted an experiment using SVM pattern recognition and regression package libsvm-3.22 [24], which has C, Java, Matlab, Python and other language versions. The experiment platform includes a CPU of Intel® Core™ i7-6500U, a RAM of 4.00 GB, and Windows 10 operating system.

4.3.2. Fisher-Markov Selector Feature Selection algorithm., Importance index θ of each feature is obtained from the training dataset by Fisher-Markov Selector feature selection algorithm, as shown in

Figure 1, 2 and 3, respectively. The feature index and the corresponding importance index θ are shown in Table 5. It can be seen that the Top10 features selected from dataset 1 and 3 are almost the same. This results show that the Fisher Markov Selector feature selection algorithm is well adapted.

Table 4. Experimental datasets.

	Training Set	Test Set
dataset 1	Slammer + Nimda	Code Red I
dataset 2	Slammer + Code Red I	Nimda
dataset 3	Nimda + Code Red I	Slammer

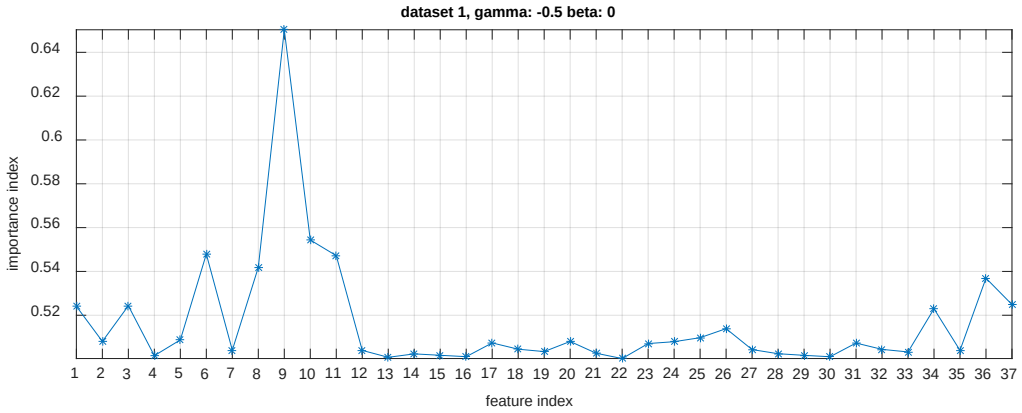

Figure 1. Importance index of features in dataset 1.

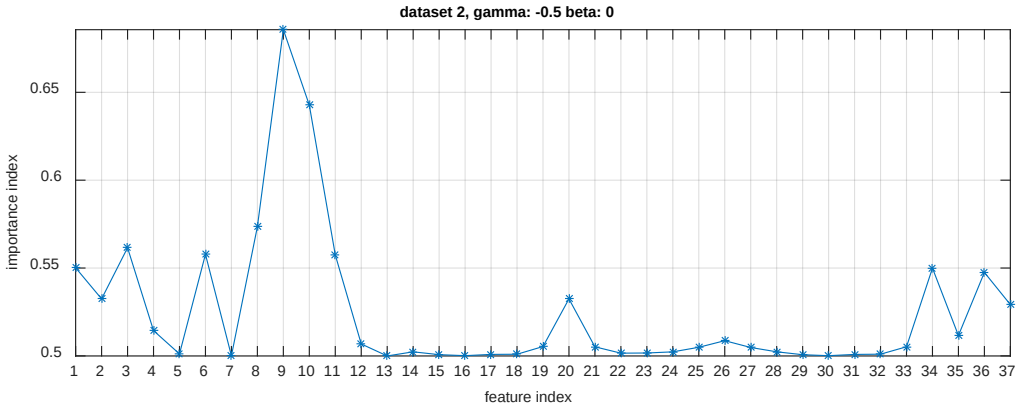

Figure 2. Importance index of features in dataset 2.

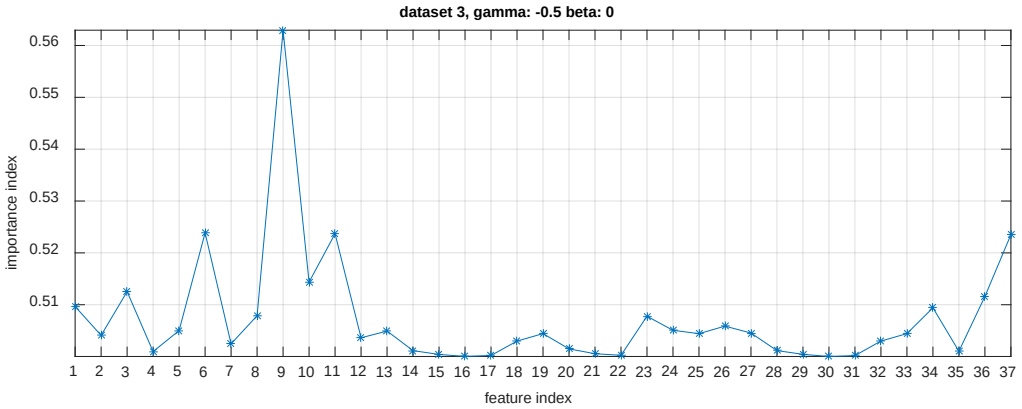

Figure 3. Importance index of features in dataset 3.

Table 5. Top10 features of importance index.

Feature Subset 1		Feature Subset 2		Feature Subset 3	
f_1	θ	f_2	θ	f_3	θ
9	0.6503	9	0.6856	9	0.5629
10	0.5544	10	0.6427	6	0.5329
6	0.5479	8	0.5735	11	0.5237
11	0.5472	3	0.5616	37	0.5235
8	0.5419	6	0.5578	10	0.5144
36	0.5369	11	0.5575	3	0.5125
37	0.5249	1	0.5502	36	0.5116
3	0.5243	34	0.5500	1	0.5097
1	0.5241	36	0.5475	34	0.5094
34	0.5230	2	0.5324	8	0.5079

4.3.3. Parameter Optimization. The results of regular class and anomaly class weight assignment for each unbalanced dataset in the experiment are shown in Table 6. The range of the grid search is shown in equation (6), namely $C \in \{2^{-5}, 2^{-4}, ..., 2^{5}\}$ and $gamma \in \{2^{-4}, 2^{-3}, ..., 2^{0}\}$. In this paper, N is 5, and the prediction accuracy of the model is evaluated by the 5-fold cross-validation. The parameter combination with the highest average accuracy is taken as the most preferred parameter combination, and the parameters of each dataset are as shown in Table 7, 8 and 9, respectively.

$$\begin{cases} \log_2 C \in [-5, 5] \\ \log_2 (gamma) \in [-4, 0] \end{cases} \tag{6}$$

Table 6. Weight assignment.

	Class	Label	Number of Samples	Weight
dataset 1	anomaly	1	4392	2.88
	regular	-1	10008	1
dataset 2	anomaly	1	1471	8.79
	regular	-1	12929	1
dataset 3	anomaly	1	4123	2.49
	regular	-1	10277	1

4.3.4. Analysis. Table 7, 8, and 9 record the parameter selection and experimental results of each model in dataset 1, 2, and 3, respectively. The training time and test time is the average CPU time of 5 training or testing. RIPE and BCNET are used to test the generalization ability of the model. From the perspective of dataset, the model based on dataset 2 is significantly lower than the dataset 1 and 3 in classification accuracy and F1-Score, and the model generalization ability is poor, so dataset 2 is not suitable for the classification model. From the perspective of feature selection, 80% of the models trained by Fisher-Markov feature selection algorithm are better than those without feature selection, which also proves that the Fisher-Markov Selector feature selection algorithm reduces the model training time while improving the classification accuracy. From the perspective of the kernel function, the RBF kernel function is generally superior to the linear, polynomial and Sigmoid kernel functions. In particular, the global optimal model SVM_5-D obtained by feature selection and RBF kernel function in dataset 3 achieves the best classification accuracy of 91.36% and F1-Score of 96.03%.

Table 7. Dataset 1 Parameter Optimization and Experimental Results.

Dataset 1			Accuracy(%)			F1-Score(%)	CPU Time(s)	
model	feature	kernel	test	RIPE	BCNET	test	train	test
SVM$_1$-A	f_1	linear	66.21	67.40	53.54	78.38	10.69	0.89
SVM$_1$-B	f_1	quadratic polynomial	88.76	94.19	81.88	93.87	66.56	1.36
SVM$_1$-C	f_1	cubic polynomial	88.39	89.69	82.78	93.72	9.41	1.44
SVM$_1$-D	f_1	RBF	90.79	98.99	90.56	95.17	4.27	1.97
SVM$_1$-E	f_1	Sigmoid	77.76	77.10	72.22	86.88	6.41	2.14
SVM$_2$-A	1-37	linear	83.50	69.80	60.49	90.34	19.78	3.06
SVM$_2$-B	1-37	quadratic polynomial	84.53	55.78	62.08	91.09	75.41	2.75
SVM$_2$-C	1-37	cubic polynomial	79.18	66.84	52.22	87.39	13.94	2.47
SVM$_2$-D	1-37	RBF	79.99	72.53	59.86	87.98	9.19	2.50
SVM$_2$-E	1-37	Sigmoid	84.50	75.46	70.20	91.80	14.22	3.72

Table 8. Dataset 2 Parameter Optimization and Experimental Results.

Dataset 2			Accuracy(%)			F1-Score(%)	CPU Time(s)	
model	feature	kernel	test	RIPE	BCNET	test	train	test
SVM$_3$-A	f_2	linear	52.50	95.44	95.14	68.11	1.34	0.47
SVM$_3$-B	f_2	quadratic polynomial	52.79	71.54	76.53	68.34	32.41	0.39
SVM$_3$-C	f_2	cubic polynomial	52.71	94.91	91.46	68.79	2.23	0.36
SVM$_3$-D	f_2	RBF	53.79	98.73	99.93	68.82	2.11	0.80
SVM$_3$-E	f_2	Sigmoid	52.13	56.89	53.33	68.94	2.25	0.66
SVM$_4$-A	1-37	linear	52.54	92.79	86.18	67.79	2.61	1.28
SVM$_4$-B	1-37	quadratic polynomial	49.44	91.68	76.39	63.80	49.34	0.72
SVM$_4$-C	1-37	cubic polynomial	52.18	94.55	86.88	67.78	3.72	0.89
SVM$_4$-D	1-37	RBF	51.08	94.05	80.07	67.62	3.86	1.02
SVM$_4$-E	1-37	Sigmoid	51.11	92.45	83.33	62.98	6.49	1.28

Table 9. Dataset 3 Parameter Optimization and Experimental Results.

Dataset 3			Accuracy(%)			F1-Score(%)	CPU Time(s)	
model	feature	kernel	test	RIPE	BCNET	test	train	test
SVM$_5$-A	f_3	linear	90.76	77.43	54.03	93.39	18.84	1.16
SVM$_5$-B	f_3	quadratic polynomial	78.00	87.69	84.09	86.08	29.13	1.34
SVM$_5$-C	f_3	cubic polynomial	90.67	66.51	50.49	94.48	34.96	1.20
SVM$_5$-D	f_3	RBF	**91.36**	97.32	90.28	**96.03**	5.58	2.71
SVM$_5$-E	f_3	Sigmoid	89.47	84.95	75.63	93.86	7.38	1.72
SVM$_6$-A	1-37	linear	82.85	78.47	59.44	89.69	25.71	5.74
SVM$_6$-B	1-37	quadratic polynomial	86.17	52.85	40.83	92.34	37.83	3.23
SVM$_6$-C	1-37	cubic polynomial	69.08	71.16	49.79	80.09	38.22	3.09
SVM$_6$-D	1-37	RBF	87.81	78.89	63.89	93.50	14.56	3.31
SVM$_6$-E	1-37	Sigmoid	89.42	82.62	75.06	93.87	18.36	4.45

5. Conclusion

In this paper, we abstract the BGP anomaly detection problem into a binary classification problem, and propose a super vector machine-based BGP anomaly detection method (SVM-BGPAD). Experimental results show that the SVM-BGPAD method can achieve the best classification accuracy of 91.36% and F1-Score of 95.03%, which is better than the existing machine learning-based BGP anomaly detection method [11-14].

The BGP update messages have a timestamp field, which can bring more information about routing state change by analyzing the time series attribute of messages. How to analyze the impact of historical BGP update messages on BGP anomaly detection in combination with time series attributes will be the focus of our next step.

Acknowledgments

This work was financially supported by the National Key Research and Development Program of China (2018YFB0803603), the National Natural Science Foundation of China (61802436, 61502531), and the Nature Science Foundation of Henan Province (162300410334).

References

[1] Y. Rekhter, T. Li and S. Hares, A border gateway protocol 4 (BGP4). No. RFC4271. 2005.
[2] MURPHY S. RFC 4272: BGP security vulnerabilities analysis. http://tools.ietf.org/html/rfc4272, 2006.
[3] B. Al-Musawi, P. Branch, and G. Armitage. BGP Anomaly Detection Techniques: A Survey[J]. IEEE Communications Surveys and Tutorials, 2017, 19(1): 377-396.
[4] A. Toonk. A BGP leak made in Canada[EB/OL]. http://www.bgpmon.net/a-bgp-leak-made-in-canada, 2012.
[5] A. Toonk. Large hijack affects reachability of high traffic destinations[EB/OL]. http://bgpmon.net/large-hijack-affects-reachability-of-high-traffic-destinations/.
[6] G. Theodoridis, O. Tsigkas, and D. Tzovaras. A novel unsupervised method for securing BGP against routing hijacks[M]//Computer and Information Sciences III. Springer, London, 2013: 21-29.
[7] X. Shi, Y. Xing, Z. Wang, and X. Yin. Detecting prefix hijackings in the internet with argus[C]//Proceedings of the 2012 Internet Measurement Conference. ACM, 2012: 15-28.
[8] Z. Zhang, Y. Zhang, and Y. Hu. iSPY: Detecting IP prefix hijacking on my own[J]. IEEE/ACM Transactions on Networking (TON), 2010, 18(6): 1815-1828.
[9] M. Cheng, Q. Xu, and J. Lv. MS-LSTM: A multi-scale LSTM model for BGP anomaly detection[C]//2016 IEEE 24th International Conference on Network Protocols (ICNP). IEEE, 2016: 1-6.
[10] L. Zhang, Y. Cui, and J. Liu. Application of machine learning in cyberspace security research[J]. Chinse Journal of Computers, 2018:1-35.
[11] N. Al-Rousan, and Lj. Trajkovic. Machine learning models for classification of BGP anomalies[C]// IEEE, International Conference on High Performance Switching and Routing. IEEE, 2013:103-108.
[12] Y. Li, H. Xiang, and Q. Hua. Classification of BGP anomalies using decision trees and fuzzy rough sets[C]//Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on. IEEE, 2014: 1312-1317.
[13] P. Batta, M. Singh, and Z. Li. Evaluation of Support Vector Machine Kernels for Detecting Network Anomalies[C]// IEEE International Symposium on Circuits and Systems. IEEE, 2018.
[14] Q. Ding, Z. Li, and P. Batta. Detecting BGP anomalies using machine learning techniques[C]// IEEE International Conference on Systems, Man, and Cybernetics. IEEE, 2017:3352-3355.
[15] Q. Cheng, H. Zhou, and J. Cheng. The fisher-markov selector: fast selecting maximally separable feature subset for multiclass classification with applications to high-dimensional data[J]. IEEE transactions on pattern analysis and machine intelligence, 2011, 33(6): 1217-1233
[16] C. Cortes, and V. Vapnik. Support-vector networks[J]. Machine learning, 1995, 20(3): 273-297.

[17] D. Moore, V. Paxson, and S. Savage. Inside the slammer worm[J]. IEEE Security & Privacy, 2003, 99(4): 33-39.
[18] A. Machie, J. Roculan, and R. Russellu. Nimda worm analysis[R]. Tech. Rep., Incident Analysis, Security Focus, 2001.
[19] D. Moore, and C. Shannon. Code-Red: a case study on the spread and victims of an Internet worm[C]//Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement. ACM, 2002: 273-284.
[20] RIPE NCC [Online]. Available: http://www.routeviews.org/. Accessed: Feb. 28, 2018
[21] S. Lally, T. Farah, and R. Gill. Collection and characterization of BCNET BGP traffic[C]//Communications, Computers and Signal Processing (PacRim), 2011 IEEE Pacific Rim Conference on. IEEE, 2011: 830-835.
[22] libBGPdump.[Online]. Available: http://bitbucket.org/ripencc wiki/Home. Accessed: Sep. 17, 2018
[23] T. Manderson. Multi-threaded routing toolkit (MRT) border gateway protocol (BGP) routing information export format with geo-location extensions. RFC 6397, IETF, Oct. 2011 [Online]. Available: https://tools.ietf.org/html/rfc6397.txt
[24] C. Chang, and C. Lin. LIBSVM: A library for support vector machines. Software available at http://www.csie.ntu.edu.tw /~cjlin/libsvm.