Network anomaly detection based on ensemble learning

Tong Pan*, Wei Chen, Long Qian
School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China

ABSTRACT

With the improvement of facility condition and network speed, the network traffic has also presented an exponential growth in recent years. However, a growing number of problems concerning cyber security have appeared. Anomaly network traffic detection can identify abnormal traffic from massive traffic. Accurate identification can reduce anomaly network traffic and protect user client. Our research is based on the traffic collected and processed by National Chiao Tung University. These data include normal traffic and abnormal traffic, the former one being majority. We propose a processing method with high accuracy. We first pre-sampled the data set, and then analysed the data features. Later on, based on theoretical research, we integrated the model of random forests and other boosting method and proposed a greedy multi-classification to binary classification model based on ensemble learning model.

Keywords: Ensemble learning, random forest, boosting algorithm, vote algorithm, network anomaly detection

1. INTRODUCTION

In recent years, the network has become an indispensable part of people's life. As the incidences of network attacks are increasing, network security is gaining more and more attention. The traditional IDS, IPS, firewall, anti-virus software and other devices have been unable to protect people's online safety. Most of these traditional devices rely on inflexible rule-based detection, and are limited in the number of attack modes which they can defend. As the threshold of network attack is getting lower and lower, the cost of hackers launching an attack is also very low, and the types and ways of network attacks are growing in number, so it is very necessary to build an intelligent traffic detection system. This research can identify attacks from the perspective of traffic and reduce the harm caused by malicious network traffic to devices.

In this paper, an ensemble learning network anomaly detection model based on Random Forest and boosting method is proposed, and anomaly traffic can be found in time. The model has a certain probability to find Advanced Persistent Threat attacks[1].

Difficulties:

- The sample size is too large and computing resources are insufficient.
- The sample features are few.

In terms of difficulty 1, the research adopts sampling tests to save computing resources so as to quickly discover data features, select representative models, and finally take all the data for training and prediction.

In terms of difficulty 2, the original features are retained as complete as possible in the experiment, and then are extracted and transformed into useful features. The main contributions of this paper are as follows:

- Combining the advantages of Bagging and Boosting ensemble learning, logistic regression is used to efficiently combine the two methods.
- Implement greedy conversion voting from multi-label classification to binary classification.

2. RELATED WORK

In the field of intrusion detection, there have been several public excellent data sets such as KDD99[2], UNSW-NB15[3], CIC-IDS2017[4], and NSL-KDD[5]. There are also a large number of methods proposed on these data sets which detection

* pt1753757728@163.com

Third International Conference on Computer Science and Communication Technology (ICCSCT 2022)
edited by Yingfa Lu, Changbo Cheng, Proc. of SPIE Vol. 12506, 125060X
 · 0277-786X · doi: 10.1117/12.2662499

accuracy has been very high. It can be seen from these data that the attack types contained in these data sets are similar, and the major categories are almost same, but the features of these data sets are very different which are difficult to be reused. In Ji, Hyunjung[6] (2013), a method based on neural network is proposed with an accuracy of 80.23% on KDD99 datasets and 67.43% on NSL-KDD datasets. In addition, Mirsky et al.[7] (2018) put forward an intrusion detection system based on deep autoencoder. Bagui, Sikha, et al.[8] (2019) use naïve Bayes and decision tress method and achieved accuracies greater than 93.32%. In Han X Y, et al.[9] (2020), a method based on provenance-based detector is proposed for advanced persistent threats.

3. SYSTEM OVERVIEW

The core idea of our model is shown in Figure 1. Firstly, the methods of Bagging and Boosting[10] in ensemble learning are applied to select models suitable for data sets. Considering that although the logistic regression model cannot achieve ideal results as expected, it performs well in processing multiple ensemble learning models, so the results of Bagging and Boosting are trained again by logistic regression, and finally the model results are obtained. In the process of training, the multi-classification is transformed into multi-sub-models of two-classification training, and the voting prediction of the output results is made based on these models. There are three key parts in the core idea: anomaly detection model part, multi-classification transformation binary classification (Section 3.2) and greedy voting (Section 3.3). Details are described below.

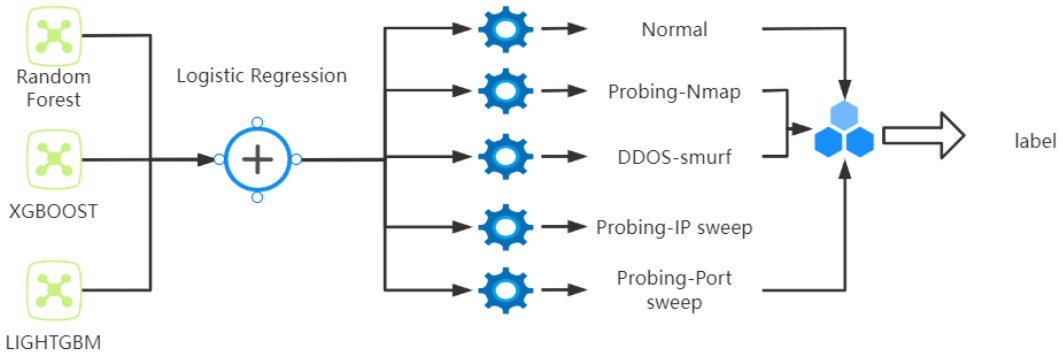

Figure 1. Train-predict model architecture diagram.

3.1 Anomaly detection based on machine learning

In related paper, deep learning and ensemble learning are used for anomaly detection. Considering that the datasets are large and the operating environment does not have enough computing resources to train the deep learning network, the paper focus on ensemble learning. The tree model has a much better classification effect by using node decision method. According to the inconsistent scope of each feature in different anomalies, it can be subdivided into multiple decision nodes to achieve good prediction results. Therefore, XGBOOST and Random Forest are mainly used for training, supplemented by LightGBM method. The following takes the random forest model as an example to illustrate in detail.

As shown in Figure 2, the original data contains address, time, protocol and other data, so it is necessary to conduct coding processing on the original data to facilitate subsequent model training, and obtain the features after coding. In addition, training models using all features probably cannot obtain excellent results, so we need to select features here, and try to incorporate features containing a large amount of information but less noise into the model. Data normalization processing is required before the beginning of training to avoid the influence of extreme values on the training model. In general, we need to divide the original data into two parts: training set and test set. The training set is used for model training, while the test set is used for verifying the model. Here is the main choice of random forest model, used in the process of training without back into the way of sampling to select feature subset from the feature space, and then build a decision tree node by splitting algorithm, is adopted for the training set data is back on the way of sampling is selected from a training set of data the subset of data, usually two-thirds of them were randomly selected randomly selected from many times, to get more training subsets, take advantage of these subsets to build decision tree, which constitute a random forest, in the end for each training sample label a stochastic model of the forest. The prediction part is to pass the test set data through the trained random forest to obtain the probability that the test data belongs to a sample tag, and the tag value corresponding to its probability value is taken as the final tag result through the voting mechanism.

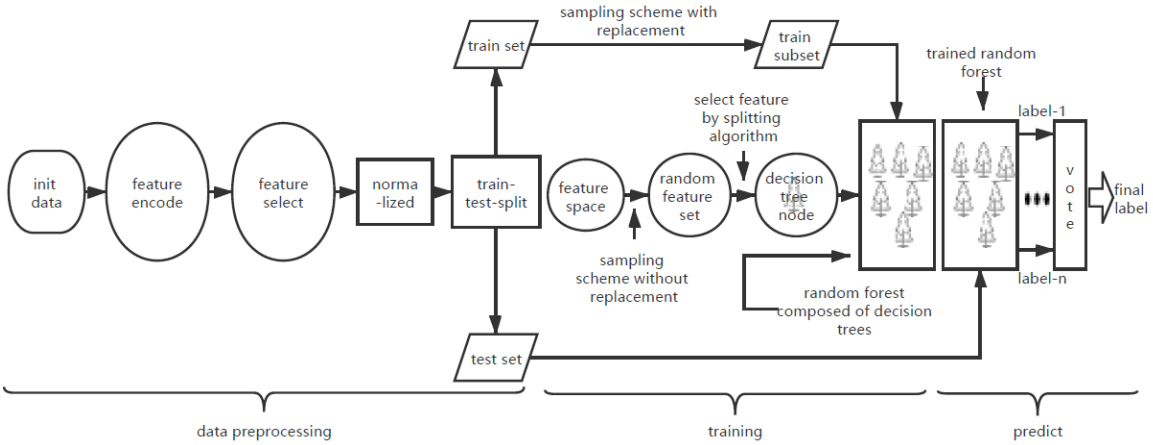

Figure 2. Anomaly detection flow chart based on random forest.

After the random forest, XGBOOST and LightGBM models are well trained, this paper proposes a framework combined with logistic regression models. Logistic regression fits the linear relationship between several models and fully extracts the characteristics of each model. However, if the accuracy of a model is poor, the overall accuracy will be inferior to that of the sub-model after logistic regression. In order to avoid overfitting, L2 regularization can be considered.

3.2 Multiple classification to binary classification

In this paper, a model is subdivided into several sub-models by transforming multi-objective classification into binary classification. The specific transformation algorithm is shown in Algorithm 3.1:

This algorithm converts multiple categories into binary categories. In consideration of efficiency, a model is randomly selected for training in line 6. If the training results are better than the existing model, the existing model is replaced.

Algorithm 3.1 Multiclass2Binary

Input: training set **Output**: multiple sub-model.

```
1:    label = select all unique label from training set;
2:    select one model;
3:    for label ≠ null do
4:        take out label [0];
5:    Label [0] class samples marked as 1, other classes samples marked as 0;
6:    training model;
7:    if trained model time > limit time then
8:    finished ← true;
9:    else
10:       save model;
11:   end for
12:   if model score > max score then:
13:       return model
14:   end
```

3.3 Multiple classification to binary classification

Different from the multi-classification to binary classification algorithm proposed in Section 3.2, the binary classification results need to be mapped back to the multiclassification. The mapping method we adopted was a voting method, and

each model was used in turn to predict the test set to get the probability of abnormal traffic. The highest probability value was taken and marked as the corresponding label. The core algorithm is shown as follows:

Integrating with the multi-classification to dichotomy algorithm proposed in Section 3.2, the binary classification results need to be remapped to multi-classification. The mapping method we adopted is greedy voting method: each model predicts the test set in turn, obtains the probability of abnormal flow, and takes the maximum probability value as the corresponding label. Greed is embodied in this on the order of the multiple binary classification model, in this paper, the two classification models according to the accuracy high and low rank, high accuracy of the model to forecast first, predict corresponding labels on two classification model, such as a sample, "A" model to predict is abnormal probability is 80%, "B" probability model prediction was 90%, but because of "A" high accuracy, the sample will be labeled "A". The core algorithm is as follows

Step 1: Use each sub-model to predict the test set and get the probability of being an exception, that is: $p(i) = \mathrm{mod}\, el_i(testset_j)$;

Step 2: Select the model with the probability greater than threshold, that is, which category should sample j be divided into, i.e.,

$$choose(j) = \arg\min_i p(i) > threshold \tag{1}$$

Step 3: Traverse all samples to get the final results.

4. EXPERIMENTS

4.1 Data processing

The main experimental environment of this paper is shown in Table 1.

Table 1. The experimental environment of the paper.

Software/Library	Version
System	Ubuntu 18.04
Jupyter	1.0.0
Python	3.6.9
Scikit-learn	0.24.0
Nmupy	1.19.4
Pandas	1.1.5

This study is based on the traffic collected and processed by NCTU[11], which has been converted into text data, and on this basis, network anomaly detection is carried out. Firstly, the number and dimension of the data are concerned. The training set contains more than 9 million samples with 22 features, and there are five labels: 'Normal', 'probing-nmap', 'ddos-smurf', 'probing-ip sweep', 'probing-port sweep'. The number of Normal samples is 892W at most, accounting for more than 90%. Secondly, we analyzed the correlation coefficient between the features of the training set, and the analysis results are shown in Figure 3. It can be seen from the figure that the maximum correlation is 0.8: CNT DST and CNT DST CONN; CNT SRC, CNT SRC SLOW, CNT SRC CONN, but considering the characteristics of the data set are relatively few, delete and merge are not carried out.

4.2 Model evaluation

This paper uses two methods to evaluate the effect of the proposed model. One is commonly used statistical indicators: macro precision, macro F1 score and macro recall, and the other is calculated by cost matrix.

The calculation equation of macro precision is as follows:

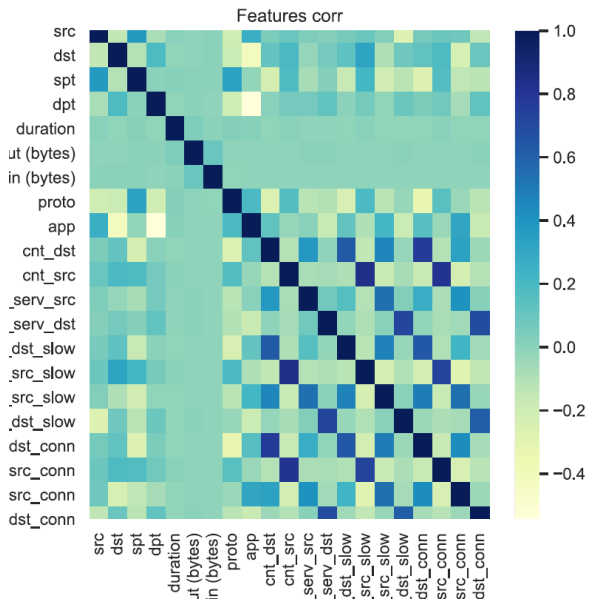

Figure 3. Heat maps of features.

$$\Pr ecision = \frac{TP}{TP + FP} \tag{2}$$

The calculation equation of macro recall is as follows:

$$recall = \frac{TP}{TP + FN} \tag{3}$$

The calculation equation of macro f1 is as follows:

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \tag{4}$$

The calculation method of the cost matrix is as follows:

$$score = \alpha * (1 - \frac{\log total\_\cos t}{\log \max\_\cos t})\beta \tag{5}$$

Among them, $\alpha = 0.3, \beta = 2$ and the cost matrix is shown in Table 2.

Table 2. Cost matrix of the model.

Cost	Normal	Probing-Nmap	Probing-Port sweep	Probing-IP sweep	DDOS-smurf
Normal	0	1	1	1	2
Probing-Nmap	2	0	1	1	2
Probing-Port sweep	2	1	0	1	2
Probing-IP sweep	2	1	1	0	2
DDOS-smurf	3	2	2	2	0

Experimental results of common models are shown in Figure 4, and those calculated by cost matrix are shown in Figure 5. It can be observed from the two figures that XGBOOST and Random Forest have good effects. Therefore, the model in this paper adopts logical regression combination of XGBOOST and Random Forest, and then adopts multi-classification greedy transformation algorithm. The model in this paper is further improved in XGBOOST and Random Forest models, with macro precision, Macro recall and Macro F1 score reaching 0.99 and 0.93 by cost matrix calculation. The model in this paper has good effect.

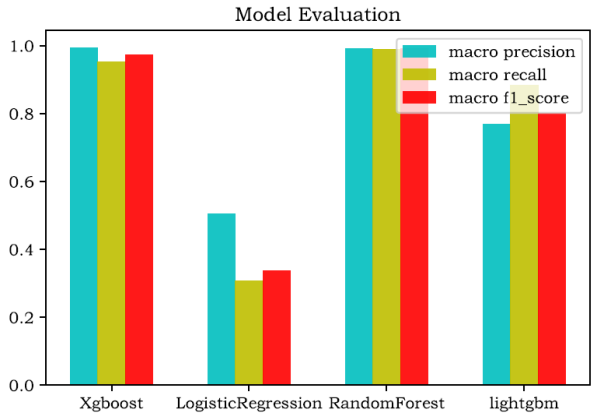

Figure 4. Statistical results of different models.

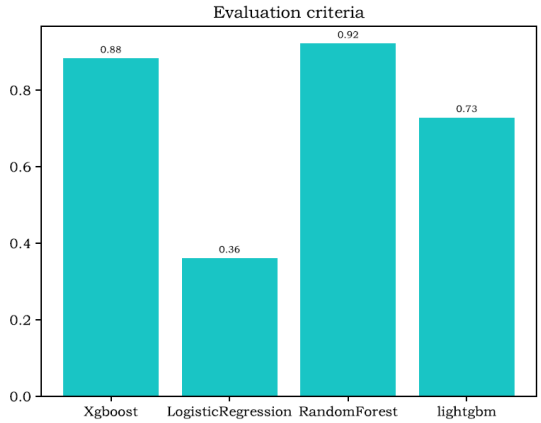

Figure 5. The result of cost matrix evaluation.

5. CONCLUSION

Based on the results and discussions presented above, the conclusions are given as below: The feature processing part of this study is a little rough, and the relationship between features may not be fully explored. Due to the large sample size and insufficient computing resources, this paper does not adopt the deep learning model. From the experimental results, the prediction accuracy is very high, but may be different from the actual flow performance. It can be seen from the experiments results that the effect of linear model is not ideal, while the detection result of tree model is very good, which is consistent with the expectation in principle. There is no strong linear relationship between feature and result, and logistic regression effect is definitely not good. However, if logistic regression is applied to model combinations, good results can be obtained. XGBOOST and Random Forest models have the best performance, with f1 of 0.99 on their respective partitioned data sets. After comprehensive voting, the results are better.

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program of China under Grant No. 2019YFB2101704.

REFERENCES

[1] Radack, S. M., “Managing information security risk: organization, mission, and information system view,” (2011).
[2] KDD99, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.
[3] Moustafa, N., and Slay, J., “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” 2015 Military Communications and Information Systems Conference (MilCIS), IEEE, (2015).
[4] IDS-2017, https://www.unb.ca/cic/datasets/ids-2017.html.
[5] Dhanabal, L., and Shantharajah, S. P., “A study on NSL-KDD dataset for intrusion detection system based on classification algorithms,” International Journal of Advanced Research in Computer and Communication Engineering, 4(6), 446-452(2015).
[6] Ji, H., et al., “A study on comparison of KDD CUP 99 and NSL-KDD using artificial neural network,” Advances in Computer Science and Ubiquitous Computing, Springer, Singapore, 452-457(2017).

[7] Mirsky, Y., et al., “Kitsune: an ensemble of autoencoders for online network intrusion detection,” arXiv preprint arXiv:1802.09089, (2018).
[8] Bagui, S., et al., “Using machine learning techniques to identify rare cyber-attacks on the UNSW-NB15 dataset,” Security and Privacy, 2(6), e91(2019).
[9] Han, X., et al., “Unicorn: Runtime provenance-based detector for advanced persistent threats,” arXiv preprint arXiv:2001.01525, (2020).
[10] Chen, T. and Guestrin, C., “Xgboost: A scalable tree boosting system,” Proceedings of the 22nd Acm SIGKDD International Conference on Knowledge Discovery and Data Mining, (2016).
[11] nad2021, https://nad2021.nctu.edu.tw.