Hindawi
Security and Communication Networks
Volume 2021, Article ID 9960822, 10 pages
https://doi.org/10.1155/2021/9960822

WILEY | Hindawi

Research Article

Malicious Encryption Traffic Detection Based on NLP

Hao Yang[1], **Qin He**[1], **Zhenyan Liu**[1] and **Qian Zhang**[2]

[1]*School of Computing Science, Chengdu University of Information Technology, Chengdu 610225, China*
[2]*School of Computer Science, University of Nottingham Jubilee Campus, Nottingham NG8 1BB, UK*

Correspondence should be addressed to Hao Yang; vhyang@foxmail.com

Received 21 March 2021; Revised 2 June 2021; Accepted 26 July 2021; Published 3 August 2021

Academic Editor: Yuan Tian

The development of Internet and network applications has brought the development of encrypted communication technology. But on this basis, malicious traffic also uses encryption to avoid traditional security protection and detection. Traditional security protection and detection methods cannot accurately detect encrypted malicious traffic. In recent years, the rise of artificial intelligence allows us to use machine learning and deep learning methods to detect encrypted malicious traffic without decryption, and the detection results are very accurate. At present, the research on malicious encrypted traffic detection mainly focuses on the characteristics' analysis of encrypted traffic and the selection of machine learning algorithms. In this paper, a method combining natural language processing and machine learning is proposed; that is, a detection method based on TF-IDF is proposed to build a detection model. In the process of data preprocessing, this method introduces the natural language processing method, namely, the TF-IDF model, to extract data information, obtain the importance of keywords, and then reconstruct the characteristics of data. The detection method based on the TF-IDF model does not need to analyze each field of the data set. Compared with the general machine learning data preprocessing method, that is, data encoding processing, the experimental results show that using natural language processing technology to preprocess data can effectively improve the accuracy of detection. Gradient boosting classifier, random forest classifier, AdaBoost classifier, and the ensemble model based on these three classifiers are, respectively, used in the construction of the later models. At the same time, CNN neural network in deep learning is also used for training, and CNN can effectively extract data information. Under the condition that the input data of the classifier and neural network are consistent, through the comparison and analysis of various methods, the accuracy of the one-dimensional convolutional network based on CNN is slightly higher than that of the classifier based on machine learning.

1. Introduction

Related principles and methods of plaintext transmission put forward higher requirements for the security of the service system, and it is an inevitable trend for Internet applications to move towards the era of comprehensive encryption [1, 2]. In recent years, the rapid increase in encrypted communications has changed the threat patterns, and many traditional methods based on conventional rules are no longer as effective as they once were. As more and more enterprises go digital, a large number of services and applications are adopting encryption as their primary means of information protection. According to NetMarketShare, the percentage of encrypted web traffic was already over 90% in October 2019. However, in the case that encrypted access can guarantee communication security, the vast majority of network devices is powerless against network attacks, malware, and other malicious encrypted traffic. A large number of malware, ransomware, proxies, remote control tools, etc., use encryption methods to avoid security protection and detection. Common security products will release unrecognized and undetectable traffic, such as Trojan, ransomware, downloaders [3], and other types of malicious software or code. In order to avoid security products and human detection, encryption is often used to disguise or hide attack behavior. Samples of malicious families that use rebound technology to bypass security devices also frequently switch back domain names and IPs and encrypt communications. The attack chain is usually divided into several steps, such as information

collection, intrusion control, achievement expansion, and battlefield cleaning. The stage of the attacker can be clearly understood through staged analysis and display of events in the traffic. Therefore, it is imperative to encrypt malicious traffic detection.

In recent years, the detection of encrypted malicious traffic has been the focus of attention in the field of network security. At present, there are two mainstream attack detection methods: detection after decryption and detection without decryption. The industry gateway devices mainly use the method of decrypting traffic to detect the attack behavior, but this solution method will consume a lot of resources, the cost is very high, also it violates the original intention of encryption, and the decryption process will be strictly limited by the laws and regulations related to privacy protection. Considering the protection of user privacy, detection methods that do not require decryption of traffic are becoming an industry of concern. Researchers are usually only allowed to observe network encrypted traffic (port 443), without decryption, by using existing data resources and standard encrypted traffic for analysis [4]. For example, the method based on statistical learning can be detected without decryption, but its detection accuracy is not high, and it cannot guarantee the correct detection of most malicious traffic. With the development of machine learning and deep learning in recent years, detection methods based on artificial intelligence have become active [5]. The methods based on machine learning or deep learning can achieve high accuracy without decryption and through some processing means.

Through many verifications, artificial intelligence used in encryption traffic security detection is a very good auxiliary means. Shengbang Security Sustainable Threat Detection and Traceability System (RayEye), based on an artificial intelligence engine, can analyze the full network traffic in real time. Combined with threat intelligence data and network behavior analysis technology, it can detect suspicious behaviors in depth, help to clearly grasp the attack chain stage and success probability of the attacker, and provide customers with malicious encryption attack traffic detection solutions.

Although the detection methods based on machine learning and deep learning do not need to decrypt the encrypted data, they still need to analyze the traffic data fields and extract the features. Based on traditional detection methods that cannot detect encryption flow and machine learning methods that need to expend energy problems such as feature extraction, this paper proposes machine learning, deep learning, and natural language processing to detect malicious traffic encryption methods, the combination uses the text classification[6] method to represent encrypted traffic, so it does not need to decrypt the data, does not need to care about the meaning of the field of the traffic data itself, and does not lose the data information of the encrypted traffic. This detection method not only is applicable to encrypted malicious traffic detection but also can be used for other related detections, such as malicious code detection. It has strong generalization and high accuracy. In the later model improvement, it is not necessary to rigidly extract the information of encrypted traffic data.

2. Materials and Methods

This paper uses the TF-IDF model to calculate the TF-IDF value of each keyword in the traffic packet and does not carry out segmentation operations on the traffic packet [7]. TF-IDF model converts qualitative data in traffic packets into quantitative data and then carries out training and detection through various classifiers. The following table shows the general situation of each classifier used in this paper.

2.1. TF-IDF. TF-IDF (Term Frequency-Inverse Document Frequency) is a normally used weighting technique for message retrieval and keyword extraction [8]. The TF-IDF model is used to calculate the TF value and IDF value of a word. If a term appears frequently in a document of a class, it is a good representation of the text of that class. Such terms should be given high weight and selected as feature terms for that class of text to distinguish it from other class documents. But only using word frequency cannot effectively filter modal words or some meaningless words. IF-IDF model introduces IDF value on the basis of word frequency. TF-IDF is a statistical approach, which is used to calculate the significance of a word to a document in a document set or a corpus. The importance of a word increases in a direct proportion with the number of times it arises in the document but decreases in an inverse proportion with the frequency of its occurrence in the corpus [9]. The main idea of TF-IDF is that if a certain word or phrase appears in one article with a high TF frequency and rarely appears in other articles, that is, IDF is low, then it is considered that this word or phrase has a good ability to distinguish categories and is suitable for classification.

TF represents word frequency, that is, the frequency of keywords appearing in the text. Vectorization of text data is to take the occurrence frequency of each word in the text as the characteristic of the text [10], and IDF value is used to correct the word frequency vector represented by TF value only.

IDF stands for reverse file frequency: its size is inversely proportional to the common degree of a word; that is, if a word is included in multiple files of a corpus, then the IDF value of the word is small. The IDF is a measure of the general importance of a word. The IDF of a particular term can be obtained by dividing the total number of files by the number of files containing the term and then taking the logarithm of the resulting quotient [11].

IDF solution formula is as follows:

$$\mathrm{IDF}(x) = \log \frac{N}{N(x)}, \tag{1}$$

where N represents the total number of Chinese texts in the corpus and $N(x)$ represents the total number of texts containing the word x in the corpus.

However, there are usually some extreme cases; for example, when a rare word does not exist in the corpus,

$N(x)$ value is 0, the above calculation formula will not be valid. Therefore, the IDF calculation formula is smoothed as follows:

$$\mathrm{IDF}(x) = \log \frac{N+1}{N(x)+1} + 1. \quad (2)$$

Therefore, the IDF value can also be correctly calculated in the above cases. Finally, the TF value and lDF value of the word are multiplied to obtain the TF-IDF value of the word. The greater the value of the word TF-IDF, the higher the importance of the word to the article. The advantage of the TF-IDF algorithm is simple and fast, and the result is more in line with the actual situation. For traffic data, TF-IDF will not cause the loss of data information and can better extract keywords to transform data information.

2.2. Detection Method. Gradient boosting belongs to the boosting series algorithm of ensemble learning [10]. Gradient boosting boosts the combination of weak classifiers. Different from the AdaBoost algorithm, gradient boosting selects the direction of gradient descent in iteration to ensure the best final result [12, 13]. Gradient boosting generates a number of weak learners, each of which takes the negative gradient as the error measurement index of the previous round of basic learners. The goal is to fit the negative gradient of the loss function of the previous cumulative model so that the cumulative model loss after adding the weak learner can be reduced in the direction of the negative gradient. Gradient boosting, compared with AdaBoost, can use any loss function (as long as the loss function is continuously differentiable), so some relatively robust loss functions can be applied, making the model more robust in noise resistance.

Random forest is a more advanced algorithm based on a decision tree (default CART tree), which also belongs to the category of ensemble learning. Random forest is composed of multiple decision trees, and decision trees do not influence each other. The random forest algorithm allows the decision tree to construct a forest randomly. After each round, each decision tree gets its own result, and the final result is determined by voting. The category with the highest number of votes is taken as the output result of the random forest [8, 14]. The introduction of randomness makes the random forest not easy to fall into overfitting. And it has good antinoise ability. Random forest can process data with high dimensions, that is, a large number of features, and it does not need to make feature selection. It has strong adaptability to data sets: it can process both discrete and continuous data, and the data sets do not need normalization.

AdaBoost algorithm belongs to boosting series of ensemble learning algorithms, which is composed of multiple weak learners and adjusts the network by giving learners different weights each time. Its adaptability lies in that the weight of the misclassified sample (the corresponding weight of the sample) of the previous weak classifier will be strengthened, and the weight of the correctly classified sample will be reduced at the same time. The sample with updated weight will be used to train the next new weak classifier again. Finally, the linear weighted sum shows that the base learner with a small error rate has a larger weight, while the base learner with a large error rate has a smaller weight [15]. In each round of training, a new weak classifier is trained with the population (sample population) to generate new sample weights and the power of the weak classifier, and the iteration continues until it reaches the predetermined error rate or the specified maximum number of iterations. AdaBoost uses exponential loss, which has a weakness that it is very sensitive to outlier points, so AdaBoost is better than gradient boosting.

Convolutional Neural Networks (CNN) [14] is a deeply structured feedforward neural network that includes convolutional computation and is mostly used in graphics processing [16]. It is one of the representative algorithms of deep learning [17, 18]. CNN usually includes data input layer, convolution calculation layer, ReLU activation layer, pooling layer, and full connection layer. CNN obtains key information mainly through continuous extraction of feature information. Compared with a machine learning algorithm, CNN can extract key information more effectively, and the number of parameters is small without careful parameter adjustment. We only need to randomly assign a weight w and a bias term b to each neuron during initialization. In the training process, these two parameters will be continuously revised to the best quality, so as to minimize the error of the model. However, this will correspondingly increase the amount of calculation, and the model training time will also increase due to the increase in the amount of calculation.

2.3. TF-IDF-Based Detection Method. Encrypted traffic messages typically contain fields such as IP address, port number, MAC address, triple handshake protocol, and various protocols. Some of these fields may directly affect the training effect of the subsequent model, while some fields may be redundant information for model training, which requires us to reserve professional network security knowledge in advance to analyze malicious traffic data, and the size, length, and field of each traffic data in the data set are different. After the encryption traffic data analysis is completed, the encrypted data are extracted by field features. The general machine learning method is to uniformly encode the input traffic data into digital form. After some feature engineering work is done on the data, the data are input to the classification model for training detection, but this method will have the problem of information loss more or less, which may make the obtained detection accuracy fail to meet the requirements. For the problem that traffic data are not easy to be processed, a detection method combining TF-IDF model in natural language processing with machine learning or deep learning is proposed. This method does not require us to consider and analyze the meaning of packets and details in specific fields, and extract relevant features of data for coding. That is to say, the TF-IDF model is used to reconstruct the data set, extract the text information of the data set, rebuild the new features, and represent the fields in

the form of vectorization when processing the data set. Using the TF-IDF model to transform traffic text will not have missing information. Moreover, better results can be obtained without further data processing. Because the data are kept in the TF-IDF model to analyze the important degree of each keyword and dealing with each keyword, instead of the need for human to deal with the analysis of data, therefore, the TF-IDF-based malicious traffic detection method can not only be used for encryption detection, but also can be applied to relevant detection that requires the use of professional technology to extract data information, such as malicious code detection.

In this paper, based on the TF-IDF model, the specific process of information text extraction and feature reconstruction for encrypted traffic data set is shown in Figure 1.

As can be seen from the figure, there are altogether 3000 pieces of encrypted traffic data in the experiment, and each piece of data is different in size, length, and field, but most of them contain text information such as transmission address, handshake information, and TCP protocol. In the experiment, a total of 906,069 keywords were obtained after the text was directly extracted from the TF-IDF model and transformed into features. The TF-IDF algorithm will calculate the TF-IDF value for each keyword. After each piece of data conversion, the original text content is converted to the corresponding keyword TF-IDF value to represent. After the conversion of the source encrypted data set, the resulting data set is a sparse matrix with the size of 3000×906069.

At the same time, the source encrypted traffic data set is transformed into digital text representation by thermal coding processing and then input into the classification model for training detection as a comparative experiment. One-Hot Encoding, also known as one-bit effective coding, mainly uses the bit status register to encode the states. Each state has its own independent register bit, and only one is effective at any time. Unique thermal coding uses 0 and 1 to represent some parameters and N-bit status register to encode N states. One-Hot Encoding can deal with discontinuous numerical features. It also expands the features to some extent.

After the encrypted traffic data set is processed by the TF-IDF model, the new data set obtained is input to each machine learning classifier and the convolutional neural network. Each classifier and convolutional neural network were trained and tested, respectively. The overall process of malicious encryption traffic detection method and experiment is shown in Figure 2.

In this paper, gradient boosting classifier, random forest classifier, and AdaBoost classifier are adopted, respectively, for training and detection, and the experimental results show that the detection results obtained by random forest classifier are better. The random forest itself is a kind of integrated learning method, and the results are obtained by voting. Therefore, based on this idea, this paper uses the ensemble learning method to combine multiple classifiers to form the ensemble model. Ensemble learning is to combine the above multiple weak supervised models in order to obtain a better and more comprehensive strong supervised model. The underlying idea of ensemble learning is that even if a weak classifier gets a wrong prediction, other weak classifiers can correct the error back. It is a meta-algorithm that combines several machine learning techniques into a prediction model to reduce bagging and increase or improve the prediction stack. In the experiment, gradient boosting classifier, random forest classifier, and AdaBoost classifier were adopted, and XGBoost classifier was combined with ensemble training. XGBoost has the characteristics of fast speed, good effect, and large-scale data processing, and XGBoost is an integrated learning framework with high accuracy, which is an efficient implementation of the GB algorithm. The final output of the ensemble learning model also adopts a voting scheme. The experimental results show that ensemble learning is better than single classifier training.

There are two options for convolutional neural networks: one is two-dimensional convolution, and the other is one-dimensional convolution based on eigenvectors [11, 19, 20]. The two-dimensional convolution first reconstructs the eigenvectors into single-channel matrices of the same size and then carries out the two-dimensional convolution. The experiment uses a variety of schemes from classical network structure (AlexNet, etc.) to specially designed network structure. However, in the course of the experiment, the following two points were found:

(1) When a simple network structure is adopted, its convergence speed is very fast [21], and it can achieve a very good effect on the training set (the accuracy rate tends to be close to 1), but the performance effect on the test set is very poor (the accuracy rate is less than 0.5), that is, the phenomenon of overfitting. After trying the processing methods including but not limited to dimension reduction, regularization, and simplified network, they cannot get better improvement.

(2) When a complex network structure is used, the convergence speed is slower, and even there are many iterations (more than 30 times), while the accuracy rate remains unchanged. The results obtained on the training set are similar to those obtained on the test set. However, the performance effect is generally poor compared with the ensemble learning scheme of machine learning (the accuracy rate is about 0.7).

Therefore, the two-dimensional convolution scheme is finally abandoned and one-dimensional convolution is adopted.

The encrypted traffic data are converted through the TF-IDF model, and the output is a sparse matrix with the size of 3000×906069. For each classifier of machine learning, including the ensemble learning model, there is no need to carry out a lot of computational fittings, so most machine learning algorithms are not limited to the size of the data set, and the general memory can meet the requirements. However, for the convolutional neural network, because each layer of the CNN network needs a lot of calculation, if

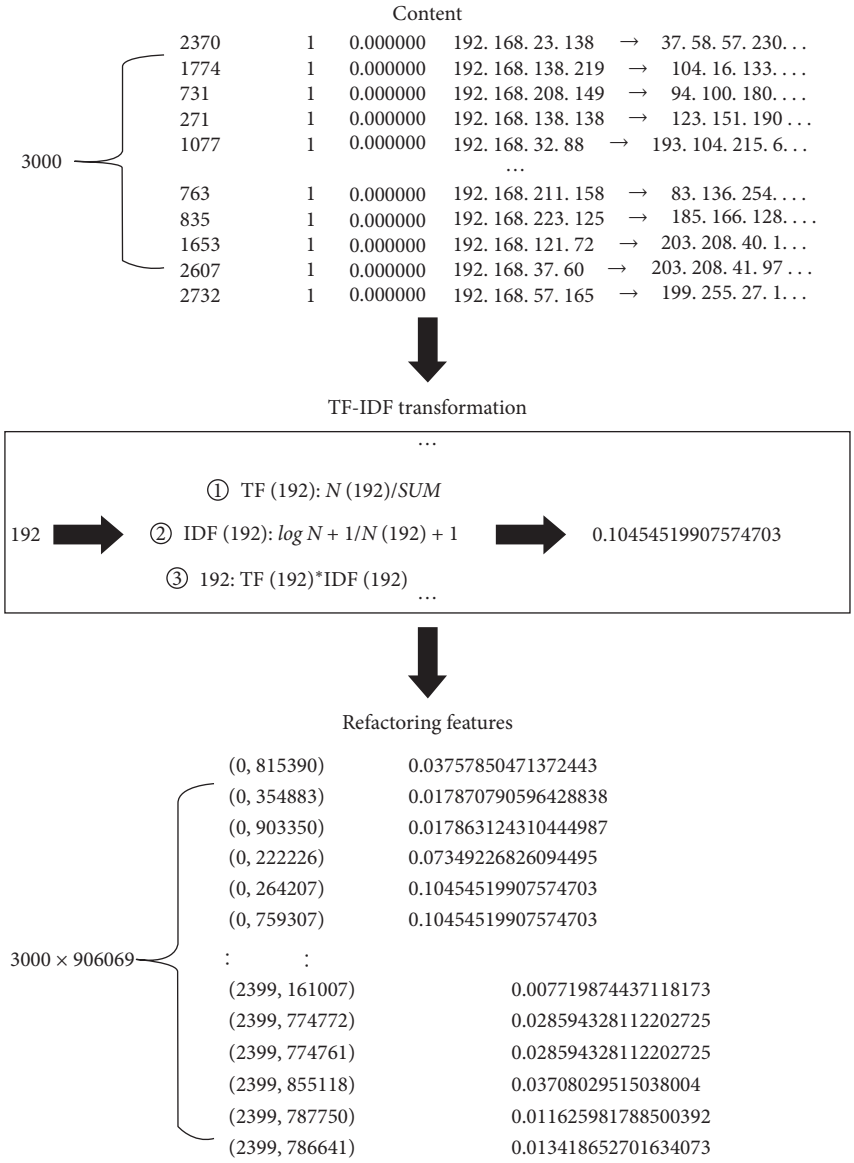

Figure 1: TF-IDF model is used to extract text information and reconstruct features from data sets.

the amount of data is too large, the general memory cannot meet the demand, and the training time will be greatly improved. Therefore, the feature dimension reduction method must be used to reduce the dimension of the data set. In this paper, Truncated SVD [22] method is used to carry out characteristic dimension reduction of data. Using Truncated SVD, the original feature matrix with size (number of texts and number of terms) is transformed into a new feature matrix with size (number of texts and number of topics). It is very suitable for data dimensionality reduction in the later stage of the TF-IDF model. The convolutional neural network structure in this paper consists of 13 layers, including the convolutional layer, the activation layer, the pooling layer, the dropout layer, the flatten layer, and the dense layer. Dropout layer is added to network results to prevent model overfitting. Add the flatten layer to convert multidimensional data to one-dimensional data. Finally, according to feature combination, the dense layer is added to classify, which greatly reduces the influence of feature position on classification.

2.4. Parameter Selection. For the ensemble learning model, it is mainly to adjust the parameters of each classifier. For the AdaBoost classifier in the ensemble model, it is mainly the decision tree classifier. Because this paper is dichotomous and the data sample is small, set max_depth to 2 and the rest to the default. Then set the maximum number of iterations of the weak learner, n_estimators, to 500. For the random forest classifier, the number of subtrees is set to 500, and the experiment shows that the effect is counterproductive when n_estimators are greater than 500. Gradient boosting classifier and XGBoost classifier are both using the default parameters. For a convolutional neural network, it is mainly about the design of iteration times and network structure. In the convolutional neural network, the convolutional layer of

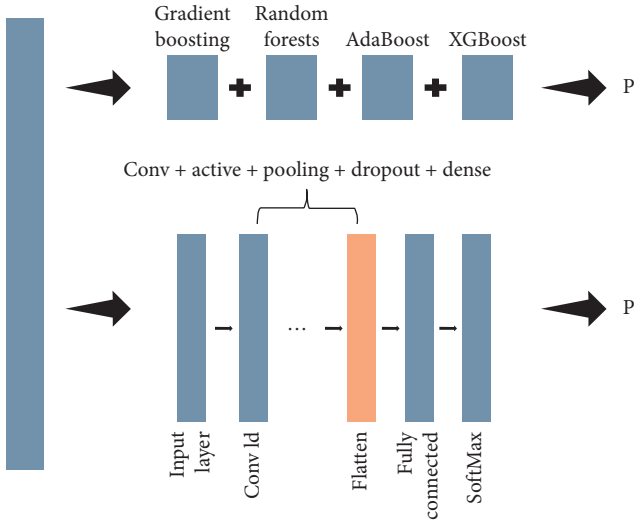

Figure 2: After the input of the training data, the results were obtained by training the data through ensemble learning and CNN, respectively.

the first few layers has a small proportion of the number of parameters, but a large proportion of the computational amount. The fully connected layer behind the network is just the opposite. Most CNN networks have this feature. Therefore, we should focus on the convolutional layer when carrying out computational acceleration optimization; when optimizing parameters and trimming weights, the focus should be on the full connection layer.

3. Results and Discussion

3.1. Data. This data set is derived from the malware and normal software collected from February to June 2020, which are operated by the sky dome sandbox of QiAnXin Technology Research Institute and filtered and generated by collecting the traffic generated. The malicious traffic defined in this data set is the encrypted traffic generated by malware (all of type exe), and the white traffic is the encrypted traffic generated by normal software (all of type exe). The traffic content is TLS/SSL packets generated by port 443. The black sample in the training set is the encrypted traffic of malware captured from February 2020 to May 2020, and the black sample in the test set is the encrypted traffic of malware captured in June 2020. All the white samples are normal software-encrypted traffic captured in 2020.

The experiment has a total of 3000 data packets, including 1500 black and white data, respectively. The experiment adopts data of 28 parts; namely, 2400 black and white data are selected as training data and 600 black and white data are selected as test data. Since the data set is a PCAP packet, the packet should be parsed first. In this experiment, Wireshark software was used to analyze the data packets. Wireshark software includes the command-line tool tshark, which can extract the desired PCAP packets by command. After analyzing PCAP data packets, data cleaning is required. Data cleaning is mainly to ensure that the collected data have a positive impact on the model. Any wrong data in the data set may have a great impact on the model construction process and the performance of the model.

When the encrypted traffic data set is large, there may be duplicate data, so we need to delete duplicate data and keep only one data. For the data set in this paper, there is basically no problem of data duplication. However, in the process of data conversion, it is inevitable that a small part of data will be wrongly copied, resulting in data duplication. This paper uses the method of matching the content of the malicious traffic field to deduplication. The parsed, deduplicated data files are then organized as DataFrame, marked with black and white labels, and shuffled out of sample order. Among them, the malicious traffic sample data are marked as 1, and the benign traffic sample data are marked as 0.

3.2. Assessment. In this paper, confusion matrix, accuracy, ROC curve, and AUC value were used to evaluate the experimental results. An obfuscation matrix is used to visually display the classification situation, and the detection results can be visually displayed for binary classification problem such as encrypted malicious traffic detection. The specific definition of the confusion matrix is shown in Table 1.

As shown in the table, TP indicates that the predicted value of the model is a benign sample, and the actual value is also a benign sample. FP indicates that the predicted value of the model is a benign sample, while the actual value is a malicious sample. FN indicates that the predicted value of the model is malicious samples, while the actual value is benign samples. TN indicates that the predicted value of the model is malicious traffic and the actual value is also malicious traffic. Accuracy, ROC curves, and AUC values are calculated on the basis of the confusion matrix.

On the basis of the confusion matrix, it can be extended to accuracy, which is our most common evaluation index, and it is easy to understand, namely, the number of samples divided by all the samples. Generally speaking, the higher the accuracy, the better the classifier. The accuracy calculation formula is as follows:

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}}, \tag{3}$$

ACC represents the proportion of all correctly judged results of the model to the total observed values, where $\mathrm{P} + \mathrm{N}$ represents the total number of use cases. TP and TN are the number of correctly classified samples.

Receiver Operating Characteristic (ROC) is a curve drawn on a two-dimensional plane, whose abscissa is the false positive rate (FPR) and the ordinate is the true positive rate (TPR). For a classifier, we can get a TPR and FPR point pair based on its performance on the test sample. Thus, this classifier can be mapped to a point on the ROC plane. By adjusting the threshold used by the classifier, we can get a curve that goes through (0, 0) and (1, 1), which is the ROC curve of the classifier. The calculation formulas of abscissa and ordinate are as follows:

Table 1: General situation of each classifier.

Data processing method	Classifier	Characteristics
TF-IDF	Gradient boosting Random forest AdaBoost	Slight time, a bit poor result
TF-IDF + SVD	Ensemble learning CNN	Long time, a bit good result

$$\mathrm{FPR} = \frac{\mathrm{FP}}{N_{\mathrm{all}}},$$
$$\mathrm{TPR} = \frac{\mathrm{TP}}{P_{\mathrm{all}}}. \tag{4}$$

In the formula, N_{all} represent the total number of negative samples and P_{all} represent the total number of positive samples. An example of ROC curve is shown in Figure 3,

Curves A and B in the figure represent the two classification models, respectively. It can be judged from the figure that the model represented by curve B performs better than model A. Meanwhile, the value of AUC is equal to the area of the graph under the ROC curve. Generally, the larger the AUC value is, the better the model effect is. In Figure 3, the AUC of model A is smaller than that of model B, so the realization effect of model B is better.

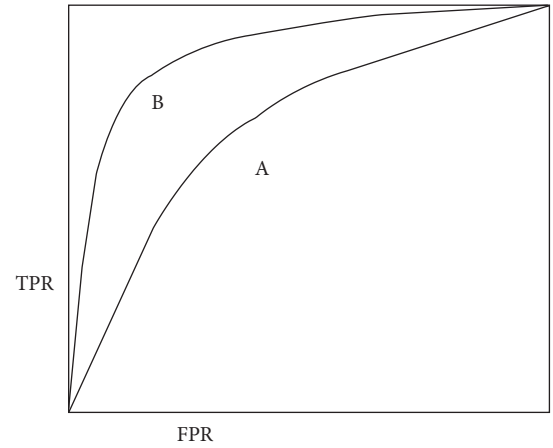

Figure 3: ROC curve diagram.

3.3. Experimental Results. After extracting keywords and reconstructing data sets from the TF-IDF model, gradient boosting classifier, random forest classifier, AdaBoost classifier, and classifier integrated with multiple classifiers are trained, respectively. Then, after the feature dimension reduction of the reconstructed data set, one-dimensional convolution CNN was input for training detection. At the same time, the source encrypted traffic data set uses One-Hot Encoding and then is input to the above classifier and CNN network training and detection. The accuracy and AUC values of the trainer and the convolutional neural network model obtained from the above experiments when testing the input test data are shown in Table 2.

It can be seen from Table 3 that, in the case of the same selected classifier and network structure, the encrypted malicious traffic detection method based on TF-IDF is significantly better than the detection method based on One-Hot Encoding. When the input data are all processed by the TF-IDF model, the detection effect of ensemble learning is better than that of other single classifiers. The detection effect of the convolutional neural network is better than that of the machine learning-based classifier, but the difference is not significant.

Table 4 shows the training time of each classifier. It can be seen from the table that the training time of the data processed by the TF-IDF model is significantly longer, and the more complex the network structure is, the more the training time is needed. Because ensemble learning is composed of many other classifiers, its training time will also increase significantly. Because of its complex network structure and a large amount of computation, the training time of CNN will also increase.

The confusion matrix obtained by TF-IDF based on the ensemble learning detection method is shown in Figure 4. After model training, 600 pieces of test data were input, among which 240 pieces of data were correctly predicted as benign samples and 317 pieces of data were correctly predicted as malicious traffic.

The ROC curve and AUC value obtained by the detection method based on TF-IDF-based ensemble learning are shown in Figure 5:

According to the ROC curve, the detection effect of the model is good, and the AUC value also reaches 0.929.

Because CNN has a better capability of feature extraction, its performance is better than ensemble learning. After the test, batch_size was set as 1000, and epoch was set as 30. In the 25th iteration, the model accuracy and loss changed little and tended to converge. Experimental results show that increasing the number of iterations will lead to a decrease in the performance of the model on the test set, and there is a tendency of slight overfitting. The changes in accuracy and loss values of its training set and test set are shown in Figure 6.

As shown in the figure, the accuracy rate of the training set tends to 1, showing a tendency of overfitting, but the accuracy rate of the test set tends to 0.933, and the effect is beyond reproach. Compared with the classifier based on machine learning, the training time of convolutional neural network is greatly increased, and it has certain requirements on the size of the data set.

Table 2: Confusion matrix definition.

		Actual category	
		True (1)	False (0)
Predicted class	True (1)	True positive (TP)	False positive (FP)
	False (0)	False negative (FN)	True negative (TN)

Table 3: The detection accuracy and AUC value obtained by different methods.

Detection method	Accuracy (TF-IDF)	Accuracy (encoding)	AUC
Gradient boosting	0.880	0.487	0.873
Random forest	0.922	0.492	0.918
AdaBoost	0.918	0.497	0.918
Ensemble learning	0.931	0.492	0.929
CNN	0.933	Huge	*

Table 4: Training time of each classifier.

Detection method	Training time (TF-IDF) (s)	Training time (encoding)
Gradient boosting	46	0.4 s
Random forest	83	6.5 s
AdaBoost	578	3.3 s
Ensemble learning	3537	11.27 s
CNN	752	Huge

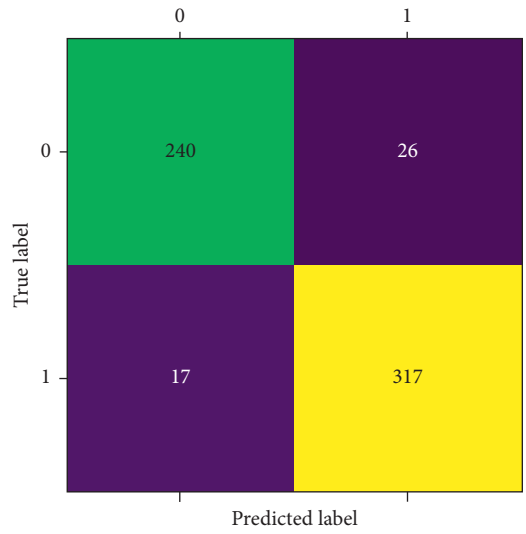

Figure 4: The confusion matrix of experimental results of ensemble learning.

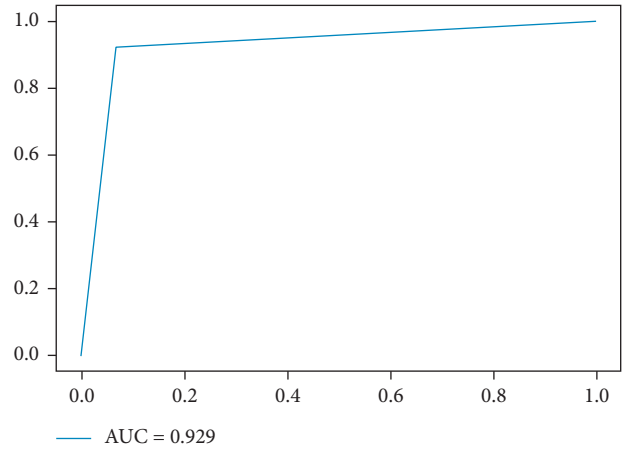

Figure 5: The ROC curve and AUC value of the experimental results of ensemble learning are obtained.

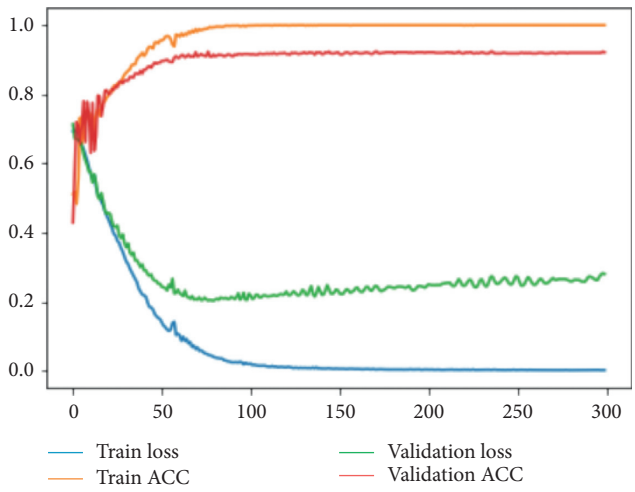

Figure 6: Loss value and accuracy value of CNN's experimental results.

4. Conclusions

Traditional detection methods, such as pattern matching, are difficult to deal with encrypted traffic data. With the development of machine learning and deep learning, the problem of encrypting data can be easily solved. At present, the detection of malicious encrypted traffic mostly adopts the method of machine learning. Although the detection method based on machine learning does not need to decrypt the encrypted traffic and is fast, it requires professionals to analyze and process the traffic data, which costs manpower and time. And the proposed detection method is based on the TF-IDF model, because the TF-IDF model does not care about the specific meaning of the data set, it replaces keywords in the source data with numbers that are calculated by their importance, so the detection method based on TF-IDF model not only applies to the malicious traffic detection but also can be used in other fields related detection, such as malicious code detection. It has strong generalization and accuracy. If the classifier model or neural network is adjusted and changed in the later stage, it is not necessary to limit the processing of the data set and information extraction. However, the detection method based on TF-IDF does not cover a comprehensive field, because the TF-IDF model simply measures the importance of a word by "word frequency," which is not comprehensive enough. Sometimes, important words may not appear many times in some data sets. In addition, the TF-IDF algorithm cannot reflect the position information of words. The words appearing in the first position and the words appearing in the second position are regarded as having the same importance, which is not accurate, and this should be taken into account in the case of different data sets.

The feature vectors reconstructed by the TF-IDF model are very sparse. This directly results in the resulting new data set being several times larger than the source data set. As the amount of data increases, memory consumption increases, leading to a significant increase in training time. For machine learning algorithms, the effect of data set size is less than that of neural network. For the convolutional neural network, due to its large amount of computation and limited by the size of the data set, there is still room for improvement in the early feature engineering processing. The experiment tried to compress the matrix, but it backfired. The following work will be improved from matrix compression, feature extraction, feature selection, and other aspects. With the further expansion of the size of the data set, if the Truncated SVD dimension reduction method cannot improve the efficiency of model construction without having a small impact on the accuracy of model recognition, then other schemes need to be reconsidered. Therefore, in the case of insufficient hardware conditions, the encrypted malicious traffic detection method based on the TF-IDF model is more suitable to use the classifier based on machine learning. Although the integrated learning model in machine learning is slightly inferior to the CNN in deep learning, the machine learning algorithm does not need to deal with the sparse matrix generated by the TF-IDF model and can retain the source data information to the maximum extent, with an accuracy rate of 0.93. Although the classifier based on ensemble learning has relatively high accuracy, the classifier based on ensemble learning has a disadvantage compared with a single classifier, which is significantly longer training time. In the case of abundant hardware resources, CNN has obvious advantages regardless of whether the sparse matrix is compressed [23–26].

Data Availability

The experimental data are real network capture packet data, provided by Qianxin Company. The data link is https://datacon.qianxin.com/opendata/maliciousstream.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was sponsored by the Sichuan Science and Technology Program (2020YFS0355 and 2020YFG0479).

References

[1] Computing Supercomputing, "Study data from Wuhan University of Technology update understanding of supercomputing (Ths-idpc: a three-stage hierarchical sampling method based on improved density peaks clustering algorithm for encrypted malicious traffic detection)," *Mathematics Week*, vol. 76, pp. 7489–7518, 2020.

[2] British Telecommunications Public Limited Company, *Patent Issued for Learned Profiles for Malicious Encrypted Network Traffic Identification (USPTO 10,594,707)*, Telecommunications Weekly, Beijing, China, 2020.

[3] O. L. Lyashuk, V. M. Klendii, O. Y. Gurik, and L. M. Slobodian, "Stand for investigation of the characteristics of screw downloaders," *Vìsnik Žitomirs'kogo Deržavnogo*

Tehnologičnogo Universitetu: Tehničnì Nauki, vol. 2, no. 80, 2017.

[4] C. Michele, D. M. Mario, L. Maurizio, and S. Andrea, "Detection of encrypted multimedia traffic through extraction and parameterization of recurrence plots. Science and engineering research center," in *Proceedings of the 2016 International Conference on Sustainable Energy, Environment and Information Engineering (SEEIE 2016)*, vol. 5, Bangkok, Thailand, March 2016.

[5] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "DISTILLER: encrypted traffic classification via multimodal multitask deep learning," *Journal of Network and Computer Applications*, vol. 183-184, Article ID 102985, 2021.

[6] A. Onan, "An ensemble scheme based on language function analysis and feature engineering for text genre classification," *Journal of Information Science*, vol. 44, no. 1, pp. 28–47, 2018.

[7] A. Onan, S. Korukoğlu, and H. Bulut, "Ensemble of keyword extraction methods and classifiers in text classification," *Expert Systems with Applications*, vol. 57, pp. 232–247, 2016.

[8] Z. Zhou, J. Qin, X. Xiang, Y. Tan, Q. Liu, and N. Xiong, "News text topic clustering optimized method based on TF-IDF algorithm on spark," *Computers, Materials & Continua*, vol. 62, no. 1, pp. 217–231, 2020.

[9] S. Yan, H. Jia, and S. Hongping, "The study of disease symptom weight mining based on text mining word frequency inverse document frequency method," *Journal of Chengdu University of Information Technology*, vol. 29, no. 1, pp. 52–58, 2014.

[10] Z. Pan, X. Yi, Y. Zhang, H. Yuan, F. L. Wang, and S. Kwong, "Frame-level bit allocation optimization based on video content characteristics for HEVC," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 16, no. 1, pp. 1–20, 2020.

[11] P. Lu and Q. Zongfeng, "Research on network public opinion detection based on improved TF-IDF algorithm," in *Proceedings of the 2019 the 9th International Workshop on Computer Science and Engineering (WCSE 2019)*, vol. 6, Hong Kong, June 2019.

[12] H. F. Jerome, "Greedy function approximation: a gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, 2001.

[13] K. Brian and B. Richard, "Small area estimation of the homeless in Los Angeles: an application of cost-sensitive stochastic gradient boosting," *Annals of Applied Statistics*, vol. 4, no. 3, 2010.

[14] J. Cheng, C. Cai, X. Tang, V. S. Sheng, and W. Guo, "A DDOS attack information fusion method based on CNN for multi-element data," *Computers, Materials & Continua*, vol. 63, no. 1, pp. 131–150, 2020.

[15] C. Guang-liang, T. Huan, Z. Fan, and Y. Sheng-liang, "AdaBoost-SVM based undergraduates evaluations," in *Proceedings of the 2019 2nd International Conference on Informatics, Control and Automation (ICA 2019)*, vol. 5, Advanced Science and Industry Research Center: Science and Engineering Research Center, Barcelona, Spain, September 2019.

[16] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, "Network traffic classifier with convolutional and recurrent neural networks for Internet of Things," *IEEE Access*, vol. 5, pp. 18042–18050, 2017.

[17] A. Onan, "Deep learning based sentiment analysis on product reviews on twitter," in *Proceedings of the International Conference on Big Data Innovations and Applications*, Springer, Istanbul, Turkey, August 2019.

[18] J. Gu, Z. Wang, J. Kuen et al., "Recent advances in convolutional neural networks," 2015, https://arxiv.org/abs/1512.07108.

[19] Z. Pan, X. Yi, Y. Zhang, B. Jeon, and S. Kwong, "Efficient in-loop filtering based on enhanced deep convolutional neural networks for HEVC," *IEEE Transactions on Image Processing*, vol. 29, pp. 5352–5366, 2020.

[20] L. Shen, X. Chen, Z. Pan, K. Fan, F. Li, and J. Lei, "No-reference stereoscopic image quality assessment based on global and local content characteristics," *Neurocomputing*, vol. 424, pp. 132–142, 2021.

[21] A. Alhussain, H. Kurdi, and L. Altoaimy, "A neural network-based trust management system for edge devices in peer-to-peer networks," *Computers, Materials & Continua*, vol. 59, no. 3, pp. 805–816, 2019.

[22] L. Shishkin Serge, A. Shalaginov, and D. Bopardikar Shaunak, "Fast approximate truncated SVD," *Numerical Linear Algebra with Applications*, vol. 26, no. 4, 2019.

[23] Z. Xiao, "Research on preprocessing method of performance monitoring data in cloud environment," in *Proceedings of the 2019 International Conference on Wireless Communication, Network and Multimedia Engineering (WCNME 2019)*, vol. 4, Advanced Science and Industry Research Center: Science and Engineering Research Center, Guilin, China, April 2019.

[24] R. Li, G. Sun, J. He et al., "Gender forecast based on the information about people who violated traffic principle," *Journal on Internet of Things*, vol. 2, no. 2, pp. 65–73, 2020.

[25] B. Mohammed and D. Naouel, "An efficient greedy traffic aware routing scheme for internet of vehicles," *Computers, Materials & Continua*, vol. 60, no. 3, pp. 959–972, 2019.

[26] A. Tahani and I. C. Alexandra, "Predicting learners' demographics characteristics deep learning ensemble architecture for learners' characteristics prediction in MOOCs," in *Proceedings of the 4th International Conference on Information and Education Innovations (ICIEI 2019)*, vol. 5, Durham, UK, July 2019.