Deep Learning-Based Encrypted Network Traffic Classification and Resource Allocation in SDN

Hao Wu[1,*], Xi Zhang[1] and Jufeng Yang[2]

[1]*School of Mechatronic Engineering and Automation, Shanghai University, Shanghai, China*
[2]*Signal and Communication Research Institute, China Academy of Railway Science Corporation, Beijing, China*
E-mail: wuhao40388949@126.com
[*]*Corresponding Author*

Received 14 July 2021; Accepted 06 August 2021;
Publication 06 November 2021

Abstract

In the rapid development of network technology, with the improvement of the quality and quantity of network users' demands, more and more network information technology and excessive network traffic also raise people's attention to the internal network security. Especially for the classification and resource allocation of encrypted network traffic, the research of related technologies has become the main research direction of the development of network technology. The extensive application of deep learning provides a new idea for the study of traffic classification. Therefore, on the basis of understanding the current situation, the improved convolutional neural network is selected to conduct an in-depth discussion on traffic classification and resource allocation of encrypted networks based on deep learning. The performance of the system is verified from the perspective of practical application.

Journal of Web Engineering, Vol. 20_8, 2319–2334.
doi: 10.13052/jwe1540-9589.2085

Keywords: Deep learning, encrypted traffic, Fourier transform, convolutional neural network, DFR architecture, one-dimensional CNN encrypted traffic classification mode.

1 Introduction

In the continuous innovation and optimization of Internet technology, in order to guarantee data transmission security and user privacy protection, network traffic began to apply encryption transmission technology. Until October 2019, the proportion of encrypted web traffic has reached 90%. Nowadays, with the increasing types of network applications and the increasing complexity of practical operation, most network application ports belong to camouflage ports or dynamic ports. On this basis, Pan W [1] et al. proposed in their experimental study that the accuracy of traffic classification based on port number could only reach 30%. Can not meet the new era of encrypted traffic classification processing needs. Therefore, researchers have put forward the detection algorithm based on the deep packet in the continuous exploration. SenS [2] and others in the study, the first use of deep packet inspection algorithm solve the problem of P2P network port mapping technology in the application, the application can use the part of the document or packet level tracking real-name signatures, and identify the relationship between the signature and application to high speed identification of P2P network traffic, improve actual operation efficiency and the accuracy of classification. With the continuous development of practical technologies, more and more researchers integrate deep packet detection and mechanical learning. For example, Xu J [3] et al. proposed to use the SAE model with deep structure to implement traffic classification during the study, and thereby obtain the payload of traffic and regard it as the input data of deep learning. Excellent achievements have been made in protocol classification, anomaly detection and unknown recognition. At the same time, Lu Yan [4] et al. converted data flow into pictures, and implemented traffic classification by using image classification model in deep learning, thus achieving the classification requirements of encrypted traffic, as shown in Figure 1 below:

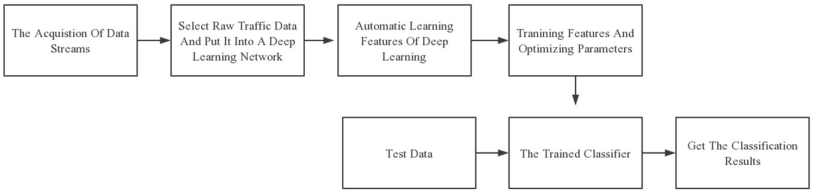

Figure 1 Flow classification and processing flow chart based on deep learning.

2 Methods

2.1 FFI – CNN Model

The convolution of the traditional convolutional neural network uses a sliding window technology to make the convolution kernel slide upward, and the convolution kernel of all the sliding regions should be multiplied and accumulated with the corresponding data. The specific formula is shown as follows:

$$d(n) * f(n) = \sum_{m=-\infty}^{\infty} d(m-n) \times f(m)dm \tag{1}$$

And improved model fast Fourier transform, convolution neural network (FFI – CNN) model, the reasonable use of convolution neural network itself advantage in classification, originally is a heart beat data L1, the length of the convolution kernels is L, if you want to make model can not restricted by the length, then the original heart take exercise unified length of data processing, And all the original data are zeroed into L2 by means of calculation [5]. The specific data presentation mode is as follows:

$$L\_2 = [(0, \ldots, 0), L\_1, (0, \ldots, 0)] \tag{2}$$

The data after the completion shall be cut into n blocks according to the fixed length L, as follows, and 5 first and tail values shall be added to each side of all data, filling the subsequent data d (n) as follows:

$$d1(n) = L\_2(nL - L + 1, nL)$$

$$d(n) = \left[\underbrace{d1}_{five}(1), d1(n), \underbrace{d1(end)}_{five} \right] \tag{3}$$

By implementing the fast Fourier transform for all data and for the convolutional kernel, multiplying the results of all data frequency fields by the convolutional nuclear frequency domain, then the product B (n) is:

$$B(n) = F(d(n)) \times F(f(n)) \tag{4}$$

To perform a fast Fourier inverse transformation of all the products, and combining the corresponding results yields the convolution definition of the study in this paper, as follows:

$$d(n) * f(n) = \sum_{n} (F^1(B(n))) \tag{5}$$

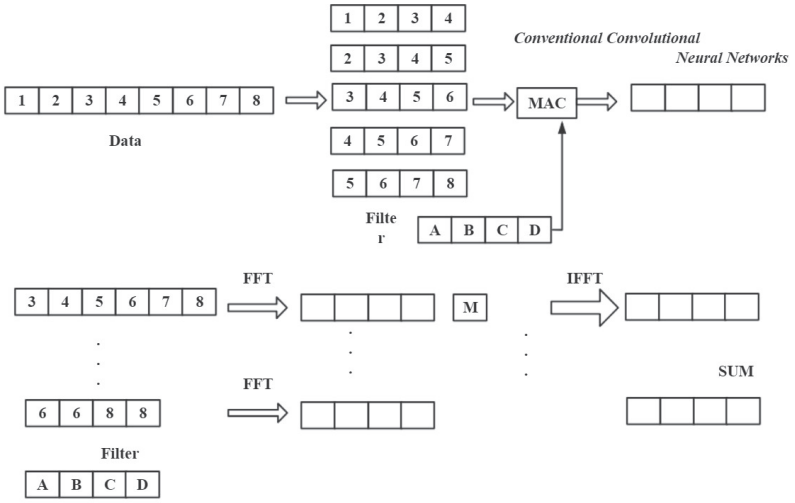

Figure 2 Comparison between traditional convolutional neural networks and the improved convolutional structure.

Combination are shown in Figure 2 below figure analysis shows that the structure of the model proposed in this paper the convolution convolution area in deep learning model let heart beat data according to the division to implement three sampling points, and then in the end, two area to extend a unit and Fourier inverse transformation, and then let the convolution kernels multiplication and the implementation of fast Fourier inverse transformation, will eventually convolution result after dealing with the pooling [6, 7]. On the basis of effectively controlling feature dimensionality reduction, then compressing parameters and synthesizing the characteristics of the full connection layer, Softmax function is used to obtain four kinds of probability values, and the network values are continuously optimized by combining back propagation. In this way, the network model can be obtained in multiple iterations, and the probability value can be used to determine the type accurately [8, 9].

FFI-CNN is similar to BP model in that it adjusts the weight value and bias by the way of back propagation, and uses gradient descent method to update in real time. The specific steps are as follows: First, complete the forward propagation of the network and define the output values of all nodes; Secondly, the loss function of this network is defined. Thirdly, the sample value of the output layer and the output residual δ are defined. Fourth, the residual δ of each node in the remaining layer is determined. Fifth, calculate the weight value w of the loss function and the partial derivative of the

offset b, as shown below; Sixth, the weight value w and offset b are updated correctly according to the gradient descent method [10, 11].

$$\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \tag{6}$$

When studying the residual transmission, we should start with the following points: First, the loss function. Relative sample set $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$.

For example, the loss functions for all samples can be defined as follows:

$$J(W, b; x, y) = -\sum_{j=1}^{T} y_i \log s_j \tag{7}$$

The above function refers to the softmax loss function, softmax layer will eventually output the probability value of all categories, and sj represents the j value of the output s, refers to the probability that this sample belongs to class j, j will range from 1 to the specific number T, can therefore regard y as a set of vectors of $1 \times$ T, including T values, only one value is 1 and the remaining T $-$ 1 values are 0. The value that belongs to 1 is the region of the real tag, and the others are 0.

Secondly, the residuals of the output layer are somewhat different from those of other layers, and the calculation formula for the residuals of all nodes I in the output layer Ni is as follows:

$$\delta_i^{(n_l)} = \frac{\partial(\omega, b; x, y)}{\partial \sigma_j^{n_l}} = -(y_i - a_i^{n_l}) \cdot f'(\sigma_j^{n_l}) \tag{8}$$

Thirdly, since the operation of the pooling layer needs to correspond all elements of the output feature layer to all regions of the convolutional layer, the residuals contained in the pooling layer should be transferred to the convolutional layer, and the specific residual calculation formula is as follows:

$$(\delta^{l+1})' = (\delta^{l+1})$$

$$\delta_i^l = \left(\sum_{j=1}^{s_{l+1}} \omega_{ji}^l (\delta_j^{l+1})' \right) f'(\sigma^l) \tag{9}$$

When calculating the residuals of the convolutional layer, the convolutional kernel transpose and the residuals after sampling should be used to

carry out the calculation, and multiplied by the derivative of the activation function. Since the structure studied in this paper is not pooled in the propagation mentioned above, the residual of the pooled layer can only be transferred to the convolutional layer. In the study of parameter update, in order to accurately obtain the above residuals and clarify the partial derivative of the loss function with respect to the weight value W and the offset b, the following formula should be used for calculation and analysis [12, 13]:

$$\frac{\partial J(\omega, b; x, y)}{\partial \omega_{ij}^{l}} = F(a_j^l)\delta^{l+1}$$

$$\frac{\partial J(\omega, b; x, y)}{\partial b_j^{l}} = \delta_i^{l+1} \tag{10}$$

For a matrix dimension of 1 × 4, the convolutional kernel dimension is 1 × 3, the output dimension is 1 × 2, and the corresponding backpropagation residue also belongs to the matrix of 1 × 2. Combined with the above analysis results, the deviation calculation formula of the weight value w is as follows:

$$\frac{\partial J(\omega, b; x, y)}{\partial \omega_{pq}^{l}} = F(a_{11}, a_{12}, a_{13}, a_{14}) * (\delta_{11}, \delta_{12}) \tag{11}$$

The fast Fourier variation of the data is as follows:

$$F(a_{11}, a_{12}, a_{13}, a_{14}) = (b_{11}, b_{12}, b_{13}, b_{14}) \tag{12}$$

Then the bias derivative of the weight value w can be represented as:

$$\begin{aligned}\frac{\partial J(\omega, b; x, y)}{\partial \omega_{pq}^{l}} &= F(a_{11}, a_{12}, a_{13}, a_{14}) * (\delta_{11}, \delta_{12}) \\ &= \sum_i \sum_j (\delta_{ij}^l b_{i+p-1, j+q-1}^{l-1})\end{aligned} \tag{13}$$

The partial derivative formula for all the convolutional kernels contained in the corresponding convolutional kernel matrix is:

$$\frac{\partial J(\omega, b; x, y)}{\partial \omega_{11}^{l}} = b_{11}\delta_{11} + b_{12}\delta_{12}$$

$$\frac{\partial J(\omega, b; x, y)}{\partial \omega_{12}^{l}} = b_{12}\delta_{11} + b_{13}\delta_{12}$$

$$\frac{\partial J(\omega, b; x, y)}{\partial \omega_{13}^{l}} = b_{13}\delta_{11} + b_{14}\delta_{12} \tag{14}$$

The bias guide of bias b, as ever, calculates the residue of layer l + 1. Combined with the weight value w and bias b bias derivative obtained from the above analysis, the gradient descent method can be updated numerical, and the specific formula is:

$$\omega_{ij}^{l} = \omega_{ij}^{l} - a\frac{\partial J(\omega, b)}{\partial \omega_{ij}^{l}}$$

$$b_{j}^{l} = b_{j}^{l} - a\frac{\partial J(\omega, b)}{\partial b_{j}^{l}} \tag{15}$$

In the above formula, it represents the j convolutional kernel corresponding to the i input of layer l, which represents the i bias of layer l, and the α represents the learning rate.

2.2 System Model

This paper takes one-dimensional CNN encrypted traffic classification as an example. The specific flow chart is shown in Figure 3, which is mainly divided into three parts: first is the flow preprocessing module; the second is the CNN model training module; third, the CNN model test module.

First, flow pretreatment. According to the method shown in Figure 4 below, traffic segmentation is required. Traffic data files should be divided into N files with only global head, packet head, n-pair packet head and packet data, so as to facilitate subsequent application. Secondly, traffic clearance redundancy and anonymous call processing should be done well. Again, we have to agree on the length. Under the condition that there is no optional field, the head of TCP is 20 bytes, while the head of UDP is 8 bytes. In order to avoid the different length directly affecting the classification result, the head of UDP is supplemented to 20 bytes in this paper. Finally, you get the image and the data set. The grayscale images converted into PNG format are made into IDX format training set and test set according to 9:1, and the two files respectively contain IDX3 and IDX1 format files. The former contains image pixels of 0–255 images and other statistical information, and the latter contains stored images and corresponding labels and statistical information.

Second, CNN model training module. The IDX3 file of the original traffic data contained in the training set and the IDX1 file corresponding to the label of the Liu table are used to train the one-dimensional CNN model, and the optimization is carried out on the basis of adjusting the parameters, and the model training is finally completed.

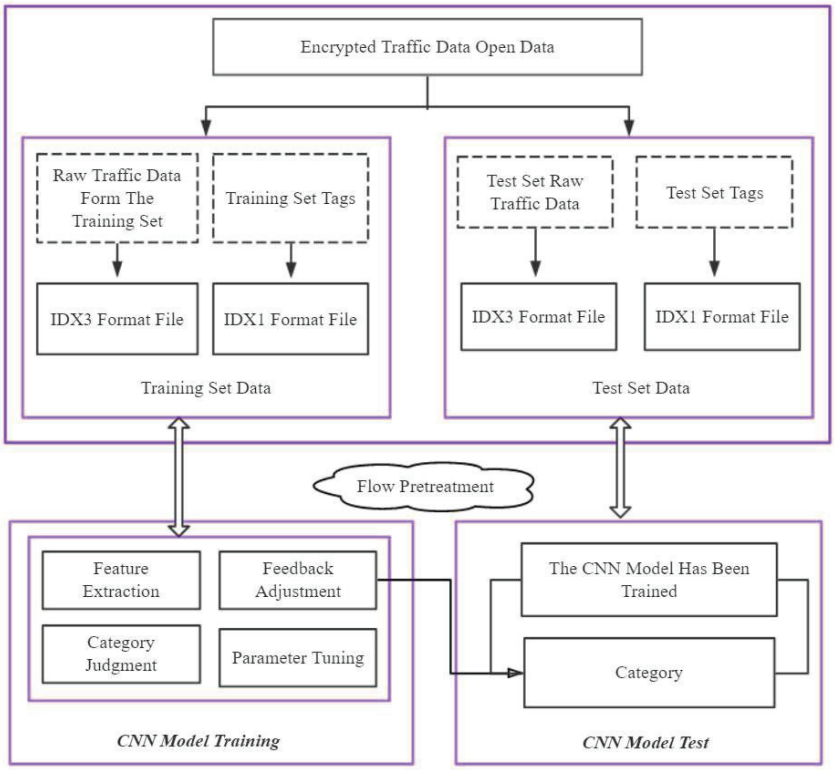

Figure 3 Is based on a one-dimensional CNN encrypted traffic classification structure diagram.

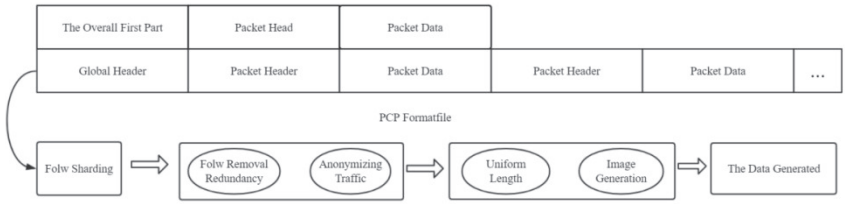

Figure 4 Preprocessing operation flow chart.

Third, CNN model test module. This module integrates the training set IDX3 and IDX1 files into the one-dimensional CNN model, and completes the overall model test after outputting the classification results.

Combined with the above system design and analysis, in the experimental test analysis, the ISCX VPN-NON VPN data set is used to disclose the

data set, so as to facilitate the accurate distinction between the pre-processed training set and the test set. This kind of data is also called VPN data set [14].

3 Results Analysis

This paper performs the performance analysis for the above outlined model, mainly using exact probability, recall and F1 values, where F1 can be seen as the average value of accuracy and recall after weighted processing with a value of 1. In the evaluation analysis, the classical F-measure is regarded as the harmonic average value of accuracy and recall, then the confusion matrix analysis based on the results of classification can obtain the following evaluation index formula:

$$\begin{aligned} A &= \frac{TP + TN}{TP + FP + TN + FN} \\ P &= \frac{TP}{TP + FP} \\ R &= \frac{TP}{TP + FN} \\ F1 &= \frac{1}{n}\sum_{1}^{n}\frac{2 * P * R}{P + R} f \end{aligned} \tag{16}$$

In the above formula, A stands for accuracy rate, P for accuracy rate, R for recall rate and F1 value. And Fi represents the ith category of encrypted traffic. Combined with the one-dimensional CNN encrypted traffic classification model proposed above, and compared with the MCLRNN model and LSTM model, it can be seen that deep learning shows more superior accuracy in encrypted traffic classification, which can meet the requirements of large-scale feature learning and classification performance, and the overall classification accuracy is also very high. Therefore, it is proved that the research on deep learning must be intensified in future technology research and development. At the same time, compared with the time consumption analysis of the three, although the CNN model is too low in the classification efficiency, the promotion of the one-dimensional CNN encrypted traffic classification model can obtain more advantages in the recall rate and accuracy. Therefore, it can be seen that this kind of technical model has a high value in practical application. In order to verify the classification of encrypted network traffic and resource storage requirements outlined in this paper, two public

Table 1 Dataset analysis analysis

Category Name	Agreement	The Number Of	The Proportion(%)
Email	SSI.&.HTTPS	9417	16.01
Chat	HTTPS	10000	17.01
Streaming	HTTPS	10000	17.01
File Transfer	HTTPS	9729	16.55
VoIP	HTTPS	10000	17.01
P2P	HTTPS	9650	16.41
A Total Of	Total	58796	100

data sets, ISCX VPN-NON VPN, are selected for framework evaluation. The specific contents are shown in Table 2 below, which is mainly used to analyze the effectiveness of encrypted traffic classification.

In experimental exploration, Tensorflow is used as simulation software and operated in the Ubuntu18.0464-bit operating system, while the selected hardware processor is 8-core Intel17-7700K CPU, internal storage capacity of 32GB,CPU accelerator belongs to two Nvidia GeForce GTX 1080Ti, and the parameters design during training analysis is shown in Table 2 below:

Table 2 Parameter design analysis

Parameter	A One-dimensional CNN	LSTM	SAE
Epoch	1600	1600	400
Minbatch	200	200	200
L.R	0.0001	0.0001	0.001
Keepp	0.5	0.5	0.5
Lambda	0.0003	0.00001	0.01
EpochFin	×	×	150000
LambdaFin	×	×	0.00002

The classification of encrypted traffic of the model constructed in this paper was evaluated and analyzed, and the causes of L1 normalization were analyzed according to the test results, and then the demand for stored data was compared and analyzed. First, by analyzing the accuracy of different classification architectures in encrypted traffic classification in Table 3 below, it can be seen that the deep learning model uses L1 regularization to replace L2 regularization, and the actual accuracy is improved by 3.29%, thus further verifying the results of the above research and analysis. At the same time, through comparative analysis, it can be seen that the higher the accuracy of one-dimensional CNN classifier is, it can reach 99.78%. Therefore, it can be

Table 3 Accuracy comparison results of classification architecture in encrypted network traffic classification

DFR Type	Accuracy (%)
L1 Regularization Based On One-dimensional CNN	99.85%
L2 Regularization Based On One-dimensional CNN	95.84%
L1 Regularization Based On LSTM	99.22%
L2 Regularization Based On LSTM	97.33%
L1 Regularization Based On SAE	98.74%
L2 Regularization Based On SAE	94.54%

regarded as a traffic classifier and applied to the current network transmission environment.

By comparison with the improved method in this paper, the analysis results as shown in Figure 5 below can be obtained. According to the analysis results of the following images, more accurate and stable classification results can be obtained, and the actual level can be increased by 12.4% and 15.2%. Recall also showed improvement over the other two algorithms. It can be seen that the framework structure proposed in this paper can scientifically classify and encrypt traffic in network transportation.

From the analysis of storage resources, the improved classification algorithm training file size reached 1,659 kB, with LSTM-core traffic classifier file of 268kB, traffic classifier training file size with SAE as the core reached 13,862 kB. Compared with the normal file size, the three have lower requirements for classification resource storage, and can be reasonably distributed in practical operation. At the same time, the one-dimensional CNN encrypted traffic classification mode proposed in this paper will regard the original network as data input, does not need to obtain features, and can implement effective preprocessing of the network traffic, remove internal excess information in time, anonymous processing, which may affect the classification results, so as to improve the time of actual model training and testing while ensuring the classification effect [15].

At the same time, when studying the throughput performance of the system, tests are implemented on the Tilera 9 nuclear platform and Tilera 36 nuclear platform. Combined with the Figure 6 above, model 3 represents the model outlined in this paper, and the performance is far higher than that of other models. In the system accuracy test analysis, the specific results are shown in Figures 7 and 8. The former refers to the accuracy test in offline state, while the latter refers to the accuracy test in online state. The comparison analysis shows, the use frequency of common applications is

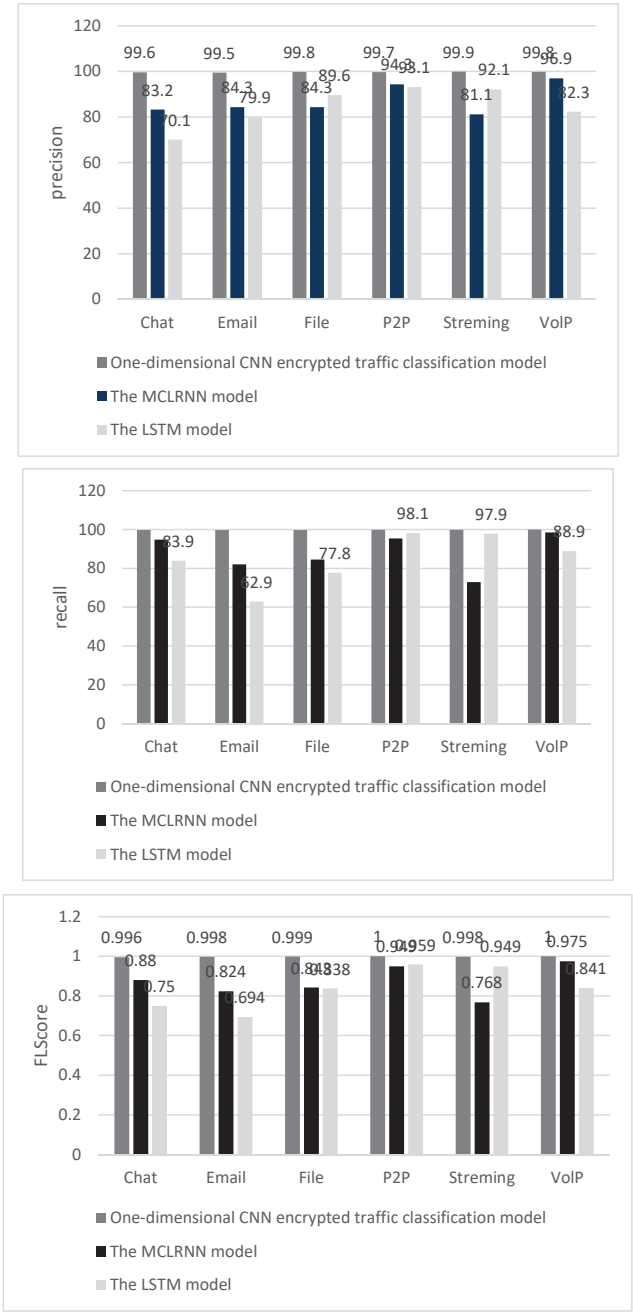

Figure 5 Comparative analysis results of encrypted traffic performance.

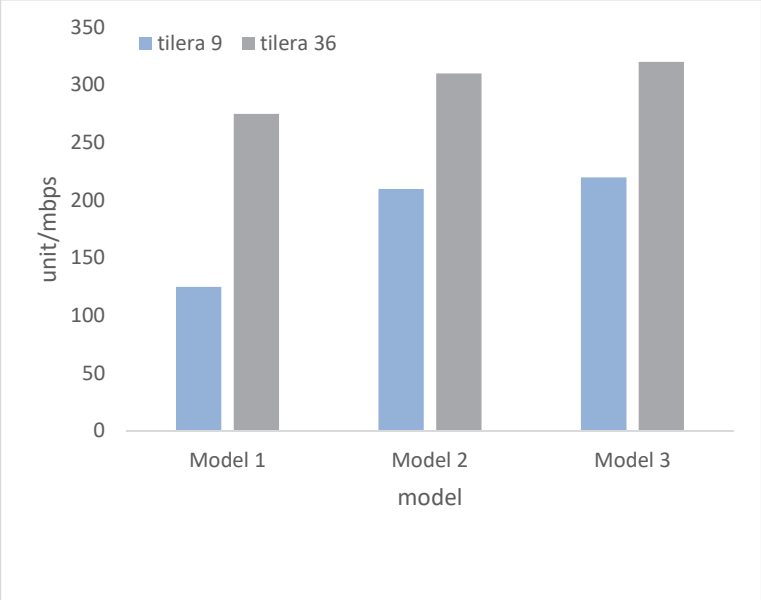

Figure 6 System throughput test and analysis results.

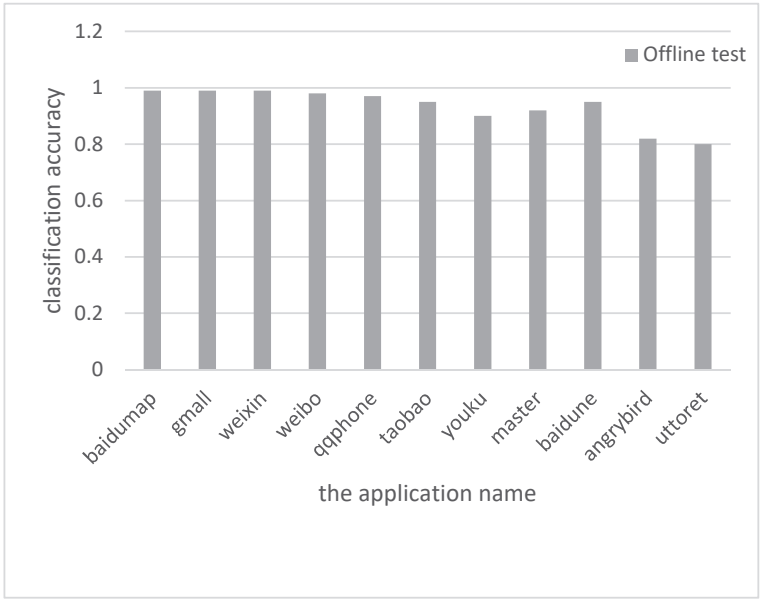

Figure 7 Accuracy test results in Offlinestate.

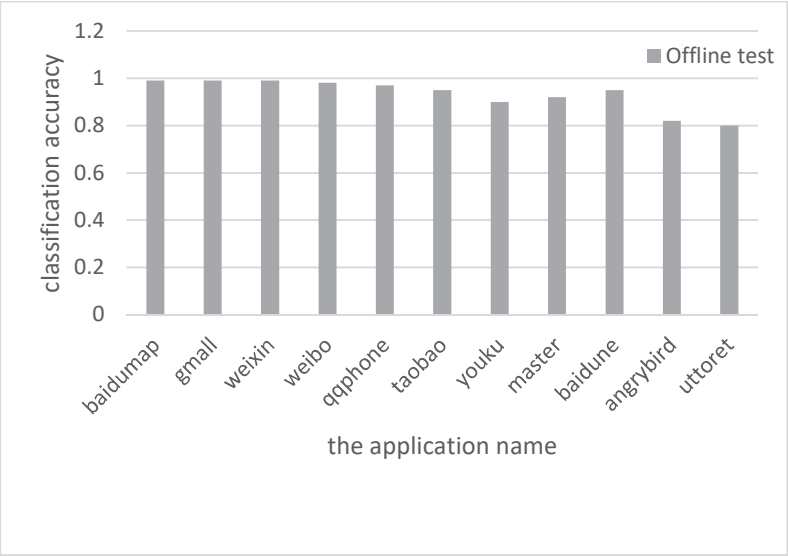

Figure 8 Accuracy test results in the online state.

high, so the test effect is excellent, generally can reach 94%, while some software with low utilization rate has low actual recognition rate.

According to the formula presented above, the accuracy, recall and F1 values of one-dimensional CNN encrypted traffic classification modes all exceeded 96.9%, compared with the existing model classification methods. Compared with other data set methods, the required parameters are also controlled, decreasing by at least 12.4% and the training time by 87.8%. It is seen that the encrypted traffic classification model of one-dimensional CNN is effective in practical applications, and can learn the characteristics of packet Ethernet, IP and TCP/UDP heads, and thus classify network traffic in an orderly manner.

4 Conclusion

In conclusion, this paper proposes an encrypted network traffic classification framework based on deep learning algorithm, verifies the application advantages of the overview framework, and performs validation analysis on representative data sets to simplify the actual resource reservation allocation. Compared with the existing machine learning methods, it can not only reduce

the operation steps of feature extraction, but also protect the privacy of users internally. The final results show that the framework can have strong encryption network traffic classification and processing technology under the condition of low requirements for storage resource traffic classification, and can ensure the actual effect of resource allocation. Therefore, in the context of the continuous improvement of future scientific research and technology, researchers should pay attention to the application value of deep learning algorithms and combine with practical exploration experience to continue in-depth research and analysis. Only in this way can they develop more high-quality technologies and accumulate more operational experience.

References

[1] Pan W, Feng Y, Chen X, et al. DataNet: Deep Learning based Encrypted Network Traffic Classification in SDN Home Gateway[J]. IEEE Access, 2018, PP(99):1–1.

[2] Sengan S, Setiawan R, Ganga R R, et al. Encrypted Network Traffic Classification and Resource Allocation with Deep Learning in Software Defined Network[J]. Wireless Personal Communications, 2021(1).

[3] Xu J, Wang J, Qi Q, et al. Deep Neural Networks for Application Awareness in SDN-based Network[C]// 2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP). IEEE, 2018.

[4] Chang L H, Lee T H, Chu H C, et al. Application-Based Online Traffic Classification with Deep Learning Models on SDN Networks[J]. Advances in Technology Innovation, 2020.

[5] Lopes F A, Santos M, Fidalgo R, et al. A Software Engineering Perspective on SDN Programmability[J]. IEEE Communications Surveys & Tutorials, 2016, 18(2):1255–1272.

[6] Niyaz Q, Sun W, Javaid A Y. A Deep Learning Based DDoS Detection System in Software-Defined Networking (SDN)[J]. Security & Safety, 2016, 4(12).

[7] Giotis K, Argyropoulos C, Androulidakis G, et al. Combining OpenFlow and sFlow for an effective and scalable anomaly detection and mitigation mechanism on SDN environments[J]. Computer Networks, 2014, 62:122–136.

[8] Tian, Shiming, Gong, et al. End-to-end encrypted network traffic classification method based on deep learning[J]. The Journal of China Universities of Posts and Telecommunications, 2020, v.27(03):25–34.

[9] Zhang C, Wang X, F Li, et al. Deep learning–based network application classification for SDN[J]. Transactions on Emerging Telecommunications Technologies, 2017:e3302.
[10] Abbasi M, Shahraki A, Taherkordi A. Deep learning for Network Traffic Monitoring and Analysis (NTMA): A survey[J]. Computer Communications, 2021(3).
[11] Guang Chen, Weiliang Han, Wenzhi Zhang., Deep Learning-Based Encrypted Traffic Classification and Intrusion Detection [J]. Computer Measurement and Control, 2020, v.28; No.256 (01): 59–65.
[12] Lee J T, Chung Y. Deep Learning-Based Vehicle Classification Using an Ensemble of Local Expert and Global Networks. IEEE, 2017:920–925.
[13] J Wan, Wu L, Xia Y, et al. Classification Method of Encrypted Traffic Based on Deep Neural Network[C]// 2019:P.54544.
[14] Hou L, Luo X Y, Wang Z Y, et al. Representation learning via a semi-supervised stacked distance autoencoder for image classification[J]. Information and Electronic Engineering Frontier: English Edition, 21(7): 14.
[15] Wubin Pan, Guang Cheng, Xiaojun Guo, et al. Review and Outlook of Network Encryption Traffic Identification Research [J]. Communications Journal, 2016, 037(009): 154–167.