Journal on Artificial Intelligence
DOI: 10.32604/jai.2022.031800
Article

Applications Classification of VPN Encryption Tunnel Based on SAE-2dCNN Model

Jie Luo*, Qingbing Ji and Lvlin Ni

Science and Technology on Communication Security Laboratory,
The 30th Research Institute of China Electronics Technology Group Corporation, Chengdu, Sichuan, China
*Corresponding Author: Jie Luo. Email: luojieanna@163.com
Received: 27 April 2022; Accepted: 20 July 2022

Abstract: How to quickly and accurately identify applications in VPN encrypted tunnels is a difficult technique. Traditional technologies such as DPI can no longer identify applications in VPN encrypted tunnel. Various VPN protocols make the feature engineering of machine learning extremely difficult. Deep learning has the advantages that feature extraction does not rely on manual labor and has a good early application in classification. This article uses deep learning technology to classify the applications of VPN encryption tunnel based on the SAE-2dCNN model. SAE can effectively reduce the dimensionality of the data, which not only improves the training efficiency of 2dCNN, but also extracts more precise features and improves accuracy. This paper uses the most common VPN encryption data in the real network to train and test the model. The test results verify the effectiveness of the SAE-2dCNN model.

Keywords: Applications classification; VPN; deep learning; SAE-2dCNN model

1 Introduction

VPN technology is often used to establish a secure tunnel on an insecure network. Data packets of different protocols are encrypted and encapsulated in the load and then transmitted through the tunnel. The VPN in this article also includes circumvention software. Due to the use of encryption technology, most VPN traffic is fully encrypted or highly disguised. Fully encrypted traffic has no clear text. The plaintext message of encrypted traffic with strong camouflage is a camouflaged HTTP header or TLS header, and the rules in it cannot be matched correctly. Therefore, technologies such as DPI and fingerprints can no longer identify the applications of VPN encrypted tunnel. Artificial intelligence has advantages that traditional manual analysis cannot surpass in traffic recognition. As early as 2016, foreign scholars identified and classified encrypted traffic based on machine learning models. Machine learning needs to process features such as packet size and timing. Feature engineering greatly affect the classification results [1]. In 2019, foreign scholars proposed a general framework for traffic classification based on deep learning. Deep learning can solve large-scale text classification problems [2]. The automatic acquisition of features by deep learning structures such as CNN can avoid complicated manual processing. SAE can reduce the dimensionality of input data and reduce

the number of CNN training rounds. This paper mainly studies how to intelligently and quickly classify applications under VPN encrypted tunnel without deciphering, proposes an application classification technology based on the SAE-2dCNN model, and inputs real network data for verification and testing.

2 Introduction to the Structure of the SAE-2dCNN Model

Autoencoder (AE) is a special kind of deep neural network without labels, which is composed of input layer, hidden layer and output layer. Data can't be reduced with general autoencoders. The use of sparse autoencoders (SAE) can inhibit some neurons to achieve the purpose of dimensionality reduction.

Convolutional neural networks can be divided into one-dimensional CNN (1dCNN), two-dimensional CNN (2dCNN) and three-dimensional CNN (3dCNN) according to the dimensions of the convolution kernel. Different CNNs need to be used in different situations. 1dCNN is suitable for classifying text and sound, 2dCNN is suitable for classifying pictures, and 3dCNN is suitable for classifying videos and 3D images. Since encrypted tunnel data can be converted into images, 2dCNN is used to classify encrypted tunnel data. Before the data is input into 2dCNN, SAE is used to reduce the dimensionality of the data, which can not only improve the training efficiency of CNN, but also learn more accurate features and improve the quality of the classifier. For one example, see Fig. 1 below [3].

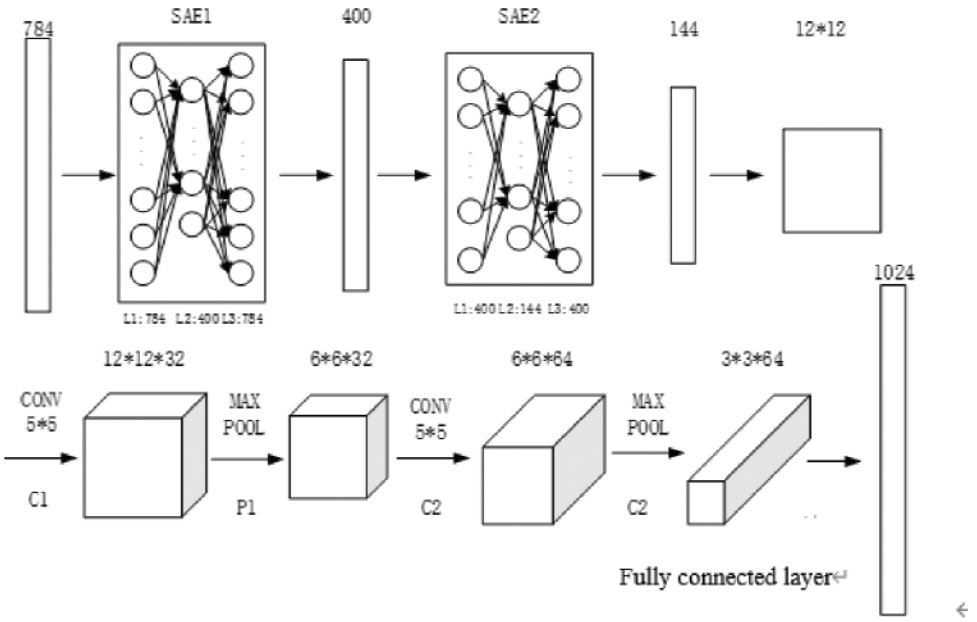

Figure 1: The structure of the SAE-2dCNN model

SAE-2dCNN uses a two-layer sparse autoencoder to extract features from the preprocessed data. The parameters of the two-layer sparse autoencoder are to be determined. The first-level autoencoder converts 784-byte pictures into 400-byte pictures through dimensionality reduction, and the second-level autoencoder converts 400-byte pictures into 144-byte pictures through dimensionality reduction. Before input to the convolutional neural network, the image is reshaped to a size of $12 * 12$.

After processing by the convolutional layer C1 with the convolution kernel size of $5 * 5$, the size of the picture is converted to $12 * 12 * 32$, which means 32 feature maps with the size of $12 * 12$. After

the $2 * 2$max pooling of the pooling layer P1, the size of the picture becomes $6 * 6$, and the number of pictures remains unchanged, so the output is $6 * 6 * 32$. After processing by the convolutional layer C2 with the convolution kernel size of $5 * 5$, the size of the picture is converted to $6 * 6 * 64$, which means 64 feature maps with the size of $6 * 6$. Similarly, after the $2 * 2$max pooling of the pooling layer P2, his size of the picture becomes $3 * 3$, and the output is $3 * 3 * 64$. After two layers of convolutional layer and pooling layer, two fully connected layers are added. The first fully connected layer converts 64 feature maps with the size of $3 * 3$ into a 1024-dimensional vector, and the 1024-dimensional vector is then converted into a vector with the 6 dimensions through the second fully connected layer, and the 6 dimensions represent the number of applications for prediction [4].

3 Classification Algorithm of SAE-2dCNN Model

Seeing Table 1 below, we design Classification algorithm of SAE-2dCNN model.

Table 1: Classification algorithm of SAE-2dCNN model

Classification algorithm of SAE-2dCNN model [5]
Randomly initialize SAE parameters W, b_{sae}, SAE hidden layer weight W', and SAE hidden layer offset b'.
Randomly initialize CNN parameters ω, b_{cnn}.
$\boldsymbol{\sigma}$ is used to calculate the universal number $\boldsymbol{L}_2$ and set the learning rate λ.
Enter VPN encrypted tunnel data $X : \left\{X_{1:n}^{(1)}, X_{1:n}^{(2)}, \ldots, X_{1:n}^{(m)}\right\}$.
Input the data X into the single hidden layer SAE, and loop the following process in each training round:
1. Calculate hidden layer output value: $h = WX + b_{sae}$
2. Calculate SAE output: $Y = W'h + b'$
3. Calculate the cost function: $J_{sparse}(W, b) = \sigma(Y - X) + \lambda W$
4. Update parameters through BP process and cost function: W', W, b', b
At the end of the loop, SAE outputs **X**, **X** is the characteristic representation of h.
Input feature representation $\boldsymbol{h}$ into CNN, and cycle the following process in each training round:
1. Calculate the feature sequence through a certain convolution kernel: $c_i = f(\omega^i \cdot \boldsymbol{h} + b_{cnn})$
2. Maximum pooling of features $c_{max} = \max\_\text{pooling}(c_1, c_2, \ldots, c_i)$
3. Through multiple sets of convolution and pooling operations, the output is obtained through the fully connected and softmax layer.
4. Through error and BP process, update ω, b_{cnn}
At the end of the loop, CNN outputs classification results.

4 SAE-2dCNN Model Training

4.1 Data Set Preparation

VPN protocols such as PPTP, IPSec, and SSH use fixed ports and standardized protocols, so they are easily blocked. Freegate, Psiphon, Unbounded and other circumvention software is unstable. Lantern and Shadowsocks (SS for short) improved the drawbacks of other circumvention software. Therefore, many domestic people who want to visit overseas websites choose Lantern and SS [6].

Due to the different protocol characteristics of different circumvention software, the data set of this model was collected separately under the most commonly used VPN software, and the model was

trained separately. The data collected are all network traffic captured in the real Internet environment, including applications based on Lantern, referred to as data set NO.1, totaling about 16 GB, of which 15G is used as the training data set and 1G is used as the test data set. The applications based on SS are referred to as the data set NO.2, totaling about 14 GB, of which 13G is used as the training data set and 1G is used as the test data set. For example, see Table 2.

Table 2: Data collection

Data set NO.1	Applications	Content	Data set NO.2	Applications	Content
Lantern 16 GB	Chat	icq, whatsapp, telegram, hangouts	SS 14 GB	Chat	icq, whatsapp, telegram, hangouts
	Email	gmail		Email	gmail
	Voip	cloudTalk, voipbuster, skype		Voip	cloudTalk, voipbuster, skype
	Streaming	vimeo, netflix, spotify, youtube		Streaming	vimeo, netflix, spotify, youtube
	Social application	facebook, twitter		Social application	facebook, twitter
	Browser	wiki, google		Browser	wiki, google

4.2 Training Data Preprocessing

1) Flow cutting: The collected flow is cut in units of flow through the software SplitCap [7].
2) Picture sample conversion: Convert the data from the data packet to the picture sample in the format of idx3 and idx1, with the following format "Fig. 2: Picture sample".

Figure 2: Picture sample

4.3 Training Data Preprocessing

The picture samples are input to SAE, and SAE reduces the dimensionality of the data. 2dCNN classifies the data after dimensionality reduction. With the following format "Fig. 3: SAE-2dCNN model training process" [8].

1) Preprocess the data to obtain training data and test data in the format of idx3, and training label files and test label files in the format of idx1.
2) Put the training data into the sparse autoencoder for unsupervised feature learning, and obtain input data with smaller dimensions, which can not only reduce the training time of each round of the convolutional neural network, but also extract the features more accurately. Finally, the classification accuracy of the classifier is increased.
3) Use the training data and training labels after dimensionality reduction to train the 2dCNN. Use SAME padding and MAX pooling methods for convolution and pooling, and use

Stochastic Gradient Descent (SGD) to minimize errors. After training, a classifier is finally used to classify the test.

4) Use the classifier trained by 2dCNN to test the test, compare the result with the test set label, and output a series of evaluation criteria to measure the classification quality of SAE-2dCNN.

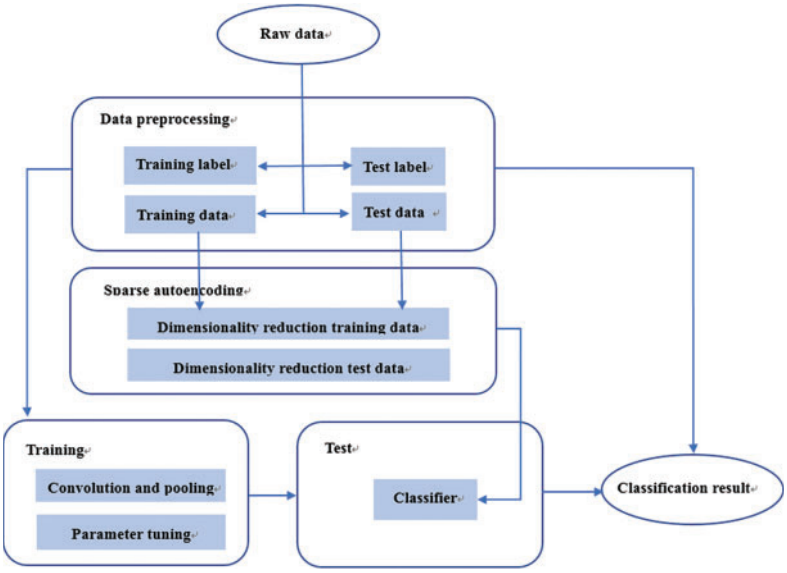

Figure 3: SAE-2dCNN model training process

5 Parameter Tuning of SAE-2dCNN Model

5.1 SAE Sparsity Parameter

Use the SAE-2dCNN model to classify the applications of VPN encrypted tunnel. Set the sparsity parameter ρ in SAE. The sparsity parameter ρ is usually a small value, which can be verified through training data, and dimensionality reduction can be better achieved when its value is determined. For example, see Figs. 4 and 5 below.

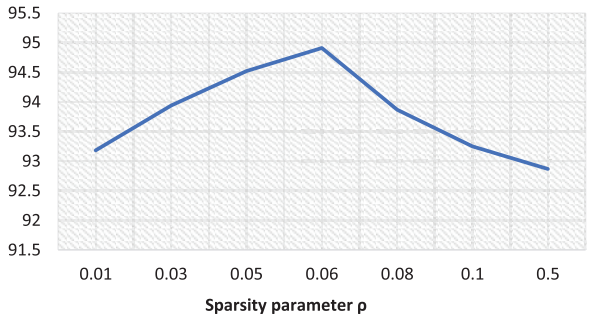

Figure 4: Sparse parameter test of the data set NO.1

Result analysis: in data set NO.1, when $\rho = 0.06$, 2dCNN can obtain the highest accuracy by inputting the reduced data. Therefore, in the SAE-2dCNN model, experiments are conducted based on the sparse parameter ρ. In data set NO.2, when $\rho = 0.05$, 2dCNN can obtain the highest accuracy by inputting the reduced data. Therefore, in the SAE-2dCNN model, experiments are conducted based on the sparse parameter ρ.

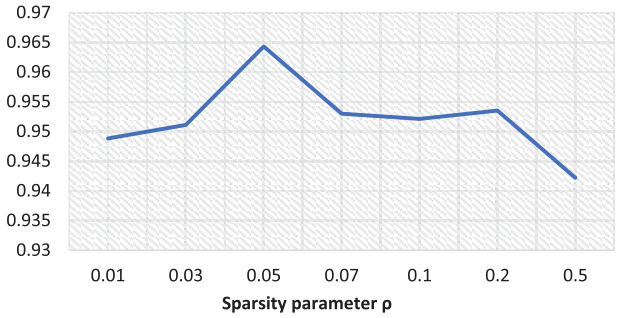

Figure 5: Sparse parameter test of the data set NO.2

5.2 SAE Sparsity Parameter

After the sparse parameters of SAE are determined, the data is input to 2dCNN for iterative loop processing, with the following format "Fig. 6: 2dCNN model training process" [9].

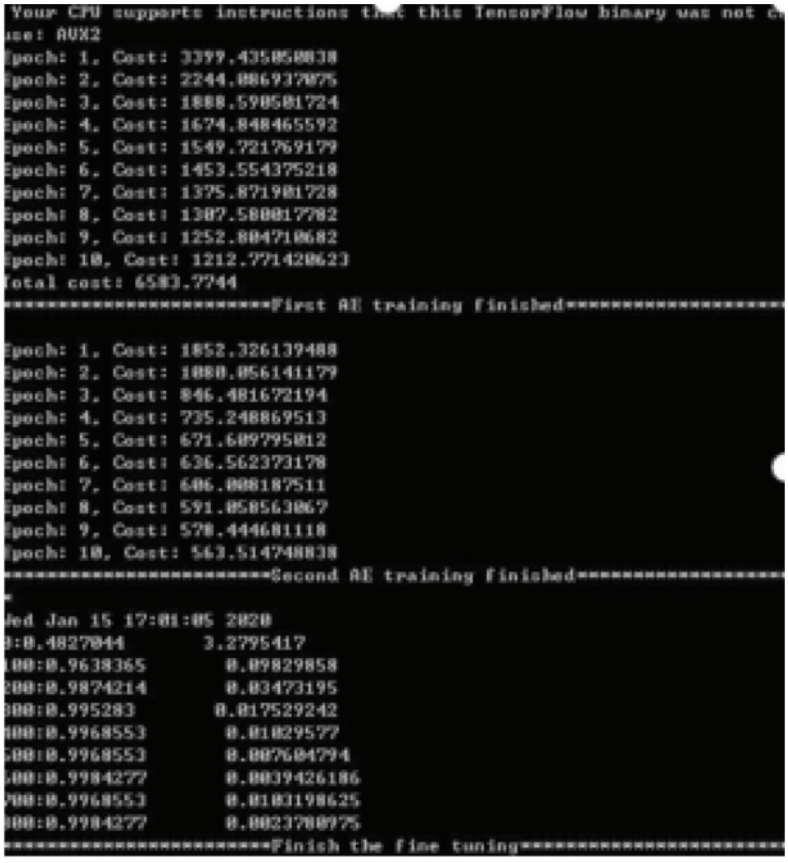

Figure 6: 2dCNN model training process

6 SAE-2dCNN Model Experimental Results and Analysis

6.1 SAE Dimensionality Reduction Results and Analysis

This article uses 2dCNN, SAE-2dCNN two models to complete the experiment on two kinds of data sets respectively. Compared with 2dCNN, this paper verifies that the SAE-2dCNN model can improve training efficiency [10].

Use 2dCNN and SAE-2dCNN to classify data set NO.1 and data set NO.2, and count the time from the start of training to the output of the results, with the following format "Fig. 7: Training time statistics of 2dCNN and SAE-2dCNN".

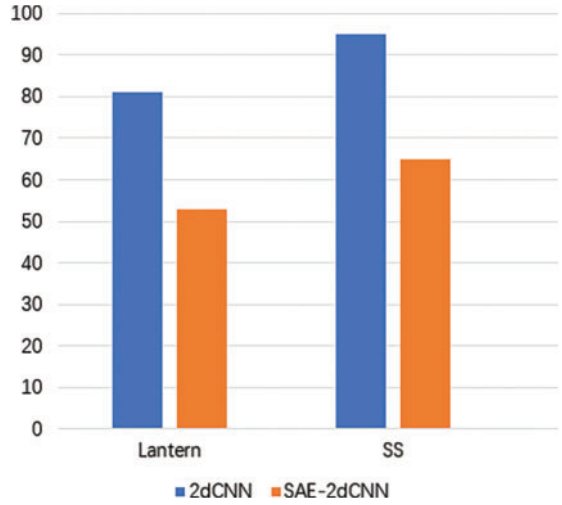

Figure 7: Training time statistics of 2dCNN and SAE-2dCNN

Result analysis: Use 2dCNN to classify the applications of data set NO.1. It takes about 81 min to train 20000 rounds of 2dCNN. It takes 53 min to train 2dCNN after adding SAE to reduce the dimensionality of the data. Excluding the 8 min training time of SAE, it takes about 45 min to train 2dCNN. Use 2dCNN to classify the applications of data set NO.2. It takes about 95 min to train 20000 rounds of 2dCNN. It takes 65 min to train 2dCNN after adding SAE to reduce the dimensionality of the data. Excluding the 12 min training time of SAE, it takes about 53 min to train 2dCNN.

It can be seen that the data after dimensionality reduction can effectively improve the training efficiency of 2dCNN.

6.2 SAE-2dCNN Model Classification Results and Analysis

Seeing Table 3 below. It shows the accuracy and recall of classification result of data set NO.1 and data set NO.2. Seeing Fig. 8 below. It is the corresponding statistical graph.

Table 3: Classification results of data set NO.1 and data set NO.2

Application	Precision (%)	Recall (%)	Application	Precision (%)	Recall (%)
L-Chat	95.77	93.57	S-Chat	80.07	89.68
L-Email	91.70	92.74	S-Email	84.34	90.24
L-Voip	94.85	97.08	S-Voip	79.04	93.56

(Continued)

Table 3: Continued

Application	Precision (%)	Recall (%)	Application	Precision (%)	Recall (%)
L-Streaming	93.07	93.55	S-Streaming	82.84	89.96
L-Social application	94.42	93.04	S-Social application	75.65	90.08
L-Browser	90.02	91.03	S-Browser	82.86	89.02

Result analysis: Comparing the classification accuracy of the data sets in Table 3, the accuracy under Lantern is about 12% higher than the accuracy under SS on average. The three applications of Chat, Voip, and Social application have higher accuracy. The application classification of Lantern is better than that of SS. Comparing the recall of the data sets in Table 3, the recall under Lantern is 2%–4% higher than the recall under SS.

It can be seen from Table 3 and Fig. 8 that from the comprehensive perspective of accuracy and recall, SAE-2dCNN has a better classification effect on data set NO.1 than data set NO.2. Because different VPN encryption tunnels use different protocols, the feature complexity is inconsistent. Lantern is an encryption protocol based on private protocol, with related features in terms of plaintext header and data packet length. SS is a proxy protocol based on camouflage, and the payload part is encrypted random with strong randomness. SS basically cannot extract features. Therefore, application classification under SS is more difficult than application classification under Lantern, and the results of SS will also be unsatisfactory.

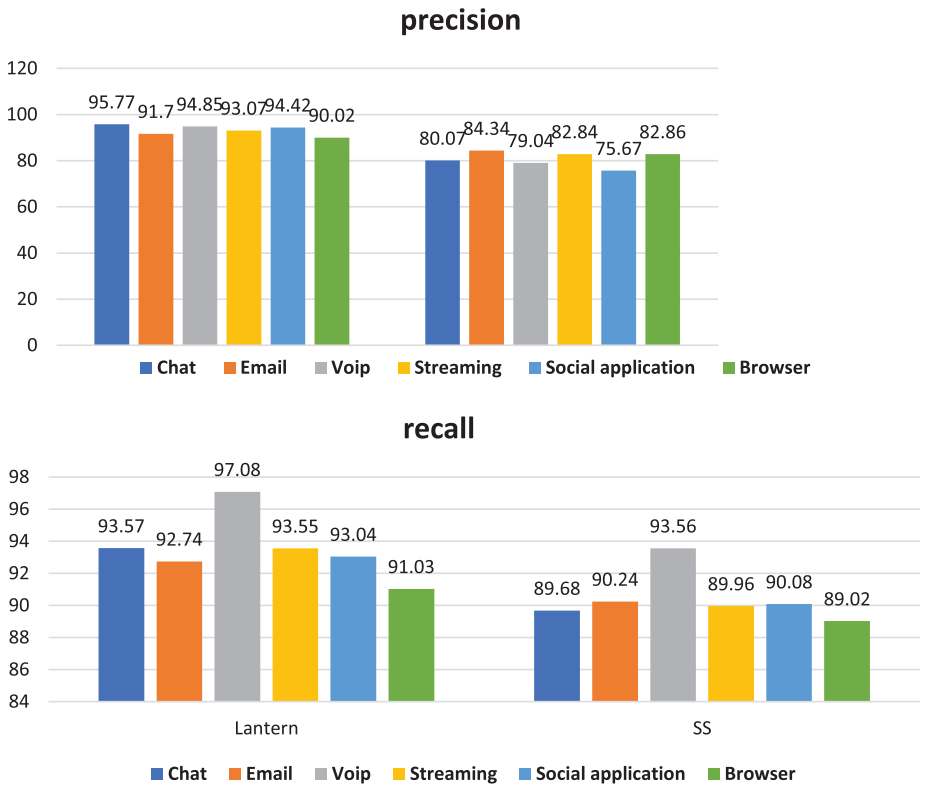

Figure 8: Applications classification of data set NO.1 and data set NO.2

Set classification evaluation value F1:

$$F1 = \frac{2 * \text{precision} + \text{recall}}{precision + reall}$$

Seeing below Table 4, Taking out the first three applications with the largest F1 values in the data sets, it can be found that: SAE-2dCNN has better classification results for the three applications of Email, Streaming and Browse. Under the same VPN encrypted tunnel, the classification result of Voip is the worst, which may be related to the complexity of Voip traffic. Therefore, if the classification effect of the application needs to be improved, manual participation is required for feature extraction.

Table 4: Classification evaluation value of data set NO.1 and data set NO.2

Application	F1	Application	F1
L-Chat	1.51	S-Chat	1.47
L-Email	1.50	S-Email	1.48
L-Voip	1.49	S-Voip	1.46
L-Streaming	1.50	S-Streaming	1.48
L-Social application	1.50	S-Social application	1.46
L-Browser	1.50	S-Browser	1.48

7 Conclusion

VPN encryption technology is a double-edged sword. On the one hand, it ensures that the encrypted information is difficult to leak, on the other hand, the network security monitoring department will face a large amount of unanalyzable data, which provides a technical means for online criminal activities. How to accurately and quickly identify applications in VPN encrypted tunnels without deciphering plays a vital role in network supervision [11].

The deep learning model has advantages in traffic recognition, which cannot be surpassed by DPI, fingerprint and machine learning. The SAE-2dCNN model proposed in this paper classifies VPN encryption tunnel applications. The most common VPN encrypted tunnel data in the real network are used to train and test the model. The test results verify the effectiveness of the SAE-2dCNN model.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. DraperGil, A. H. Lashkari, M. S. I. Mamun and A. A. Ghorbani, "Characterization of encrypted and vpn traffic using time related," in *Proc. of the 2nd Int. Conf. on Information Systems Security and Privacy (ICISSP)*, Rome, Italy, pp. 407–414, 2016.

[2] S. Rezaei and X. Liu, "Deep learning for encrypted traffic classification: An overview," *IEEE Communications Magazine*, vol. 57, no. 5, pp. 76–81, 2019.

[3] W. Wang, M. Zhu, X. Zeng, X. Ye and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *Int. Conf. on Information Networking (ICOIN)*, Da Nang, Vietnam, pp. 1145–1150. 2017.
[4] W. Wang, M. Zhu, J. Wang, X. Zeng and Y. Zhongzhen, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," in *2017 IEEE Int. Conf. on Intelligence and Security Informatics (ISI)*, Beijing, China, pp. 875–885, 2017.
[5] J. Xu, X. Chen, B. Wang and J. Cui, "A deep learning network traffic classification method," 2016.
[6] J. Y. Zhang, "Research on shadowsocks anonymous traffic identification technology based on website fingerprint," TianJin: University of Technology. 2017.
[7] T. Len, "A survey of encrypted traffic classification based on deep learning," *Computer and Modernization*, vol. 3, no. 8, pp. 112–120, 2021.
[8] R. Ma, "Research and implementation of unknown and encrypted traffic recognition based on convolutional neural network," Beijing University of Posts and Telecommunications, 2017.
[9] M. Mondaeev, T. Anker and Y. Meyouhas, "Method and apparatus for deep packet inspection for network intrusion detection," vol. 156, no. 1, pp. 163–193, US9973430B2, 2018.
[10] T. Auld, A. W. Moore and S. F. Gull, "Bayesian neural networks for internet traffic classification," *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 223–239, 2007.
[11] L. Guo, Q. Wu and S. Liu, "Deep learning-based real-time VPN encrypted traffic identification methods," *Journal of Real-Time Image Processing*, vol. 17, pp. 103–114, 2020.