IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 6 June 2023, accepted 19 June 2023, date of publication 26 June 2023, date of current version 3 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3289200

RESEARCH ARTICLE

Abnormal Traffic Detection Based on Attention and Big Step Convolution

DAOQUAN LI, XUEQING DONG, JIE GAO, AND KEYONG HU
School of Information and Control Engineering, Qingdao University of Technology, Qingdao, Shandong 266520, China
Corresponding author: Keyong Hu (hukeyong@qut.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61902205.

ABSTRACT Abnormal traffic detection is critical to network security and quality of service. However, the similarity of features and the single dimension of the detection model cause great difficulties for abnormal traffic detection, and thus a big-step convolutional neural network traffic detection model based on the attention mechanism is proposed. Firstly, the network traffic characteristics are analyzed and the raw traffic is preprocessed and mapped into a two-dimensional grayscale image. Then, multi-channel grayscale images are generated by histogram equalization, and an attention mechanism is introduced to assign different weights to traffic features to enhance local features. Finally, pooling-free convolutional neural networks are combined to extract traffic features of different depths, thus improving the defects such as local feature omission and overfitting in convolutional neural networks. The simulation experiment was carried out in a balanced public data set and an actual data set. Using the commonly used algorithm SVM as a baseline, the proposed model is compared with ANN, CNN, RF, Bayes and two latest models. Experimentally, the accuracy rate with multiple classifications is 99.5%. The proposed model has the best anomaly detection. And the proposed method outperforms other models in precision, recall, and F1. It is demonstrated that the model is not only efficient in detection, but also robust and robust to different complex environments.

INDEX TERMS Attention, histogram equalization, big step CNN, abnormal traffic detection.

I. INTRODUCTION

Internet technology is widely used in all walks of life, and has strongly contributed to the development of economy and society. However, as the current mainstream network security and defense technologies still have many shortcomings, the huge application requirements also make the security configuration of the entire network becomes particularly complex, resulting in the entire network facing the threat of extremely vulnerable to attacks. At the same time, due to the openness of the TCP/IP network architecture, computer viruses spread more widely through disguise, which affects the normal operation of the network and causes social and economic downturn. How to take effective methods to analyze data information to predict the current network development, find abnormalities and take appropriate handling measures is of great significance to maintain network security [1].

The associate editor coordinating the review of this manuscript and approving it for publication was S. K. Hafizul Islam.

Anomalous traffic detection can be achieved with the help of network traffic classification. According to its core idea there are mainly the following approaches: port-based [2], deep packet detection based [3], and machine learning based [4]. Machine learning consists of traditional machine learning and deep learning. In the early days, when the Internet was small and the traffic types were simple, the first two methods had stable performance and achieved good classification results [5], [6], [7]. However, with the continuous emergence of new Internet applications, traffic types are increasing and traffic components are becoming more complex, which reduces the classification effect. Machine learning improvement methods are proposed to address the limitations of the above methods. Machine learning is to extract statistical features of network traffic and classify them with reliable efficiency and high accuracy. It also has a wide range of application prospects.

The overall process of network traffic classification consists of collecting data sets, generating normalized data,

data pre-processing, feature extraction, training models and classification. Traditional machine learning classification is based on different algorithms to select the optimal subset of features that are similar to the full feature results for classification. This approach relies on feature selection, which can directly affect the classification results and cannot cope with the evolution of modern network traffic. And traditional machine learning models cannot represent the complex relationships between individual features. As a result, deep learning becomes the optimal algorithm for solving network traffic classification, which performs high performance in dynamic and challenging traffic classification environments.

In the past few years, deep learning has had several studies working on network traffic classification [8], [9], [10]. These studies provide the performance-enhancing feasibility of deep learning techniques for handling traffic classification tasks, but reveal that deep learning is still in its infancy for network anomaly detection research. In contrast to machine learning, deep learning not only enables classification of network traffic by automatically extracting structured and complex features and feeding them directly into a training classifier, but also represents complex nonlinear relationships between features. In summary, the anomalous traffic detection model for network security defense has improved in terms of improvement and practicality. However, there are still many problems: First, the classification results are poor for traffic information with similar attribute characteristics. Second, the structure of the anomaly network detection model is inflexible and cannot extract features in multiple dimensions and fields of view, which reduces the accuracy of network traffic classification to a certain extent. Third, multiple pooling using convolutional neural networks has the potential for information loss, which can make the sequence less relevant.

To overcome the above challenges and difficulties, this paper makes the following contributions:

- In this paper, we propose an Attention and Big Step Convolutional Neural Network (ABS-CNN) model based on the attention mechanism [11]. To solve the problems such as similar features leading to worse classification results, the attention mechanism is invited to assign attention weights to data sequences to distinguish subtle features. To solve the problems such as similar features leading to worse classification results, the attention mechanism is invited to assign attention weights to data sequences to distinguish subtle features. Experiments show that the model with enhanced features has higher classification accuracy and better robustness.
- In this paper, we use histogram equalization to solve the problem of single model dimensionality. The traffic data is first processed into grayscale images and then the images are histogram equalized. Combined with improved multi-channel convolution to automatically extract and fuse multi-field fine-grained features. The experiments show that the traffic with histogram equalization performed is relatively well-defined, which results in better model detection performance and better robustness.
- To address the reduced correlation of traffic sequences due to pooling, the traffic features are extracted by combining big-step convolution. And big-step convolution is also called stepwise convolution. Stepwise convolution preserves the sequence-related features extracted by the convolution layer and reduces the harm of accuracy loss due to information loss.

This paper is divided into five parts. Section I briefly describes the research background and main contributions of this paper. Section II will describe and summarize the current research development. Section III carries out the model introduction and algorithm implementation process. Detailed experiments and analysis of the results will be carried out in Section IV. Finally, Section V will analyze and summarize the model proposed in this paper and point out some possible future research directions.

II. RELATED WORK

Anomalous traffic detection can be achieved with the help of network traffic classification. This section presents work related to the classification of network traffic based on traditional machine learning and deep learning.

A. TRADITIONAL MACHINE LEARNING

Traditional machine learning contains algorithms such as random forests [12], decision trees [13], support vector machines [14], and naive bayes [15]. Most scholars use feature preference or improved model approach to achieve the classification of network traffic. Shi et al. [16] proposed a cost-sensitive SVM (CMSVM) for the network traffic imbalance problem. The model uses a multi-class SVM with an active learning algorithm to solve the imbalance problem for different applications by adaptive weights. Cao et al. [17] proposed a real-time network classification model with SPP-SVM. The model uses the feature selection method of principal component analysis (PCA) to reduce the dimensionality of the original data and uses an improved particle swarm optimization algorithm to obtain the optimal parameters. The classification accuracy is higher compared to the traditional SVM model. Farid et al. [18] combined naive bayes and decision trees for anomalous traffic detection while eliminating redundant attributes of the traffic data. The proposed algorithm improves the detection rate. Machine learning based classification methods usually require manual feature design and selection, which cannot cope with the evolution of networks nowadays.

B. DEEP LEARNING

The automatic feature extraction ability and versatility of deep learning make up for the shortcomings of traditional machine learning methods. Gianni et al. [19] proposed a novel deep neural network based on autoencoder. The model embeds multiple autoencoders into convolutional and

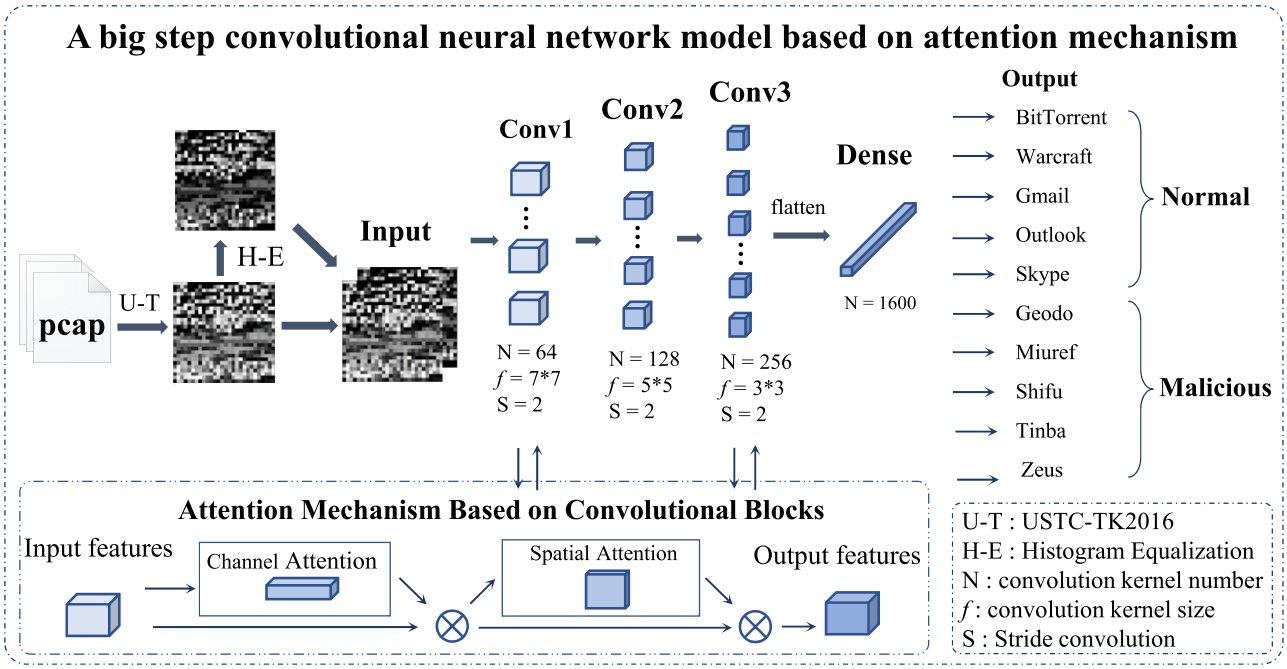

FIGURE 1. Flowchart of network traffic classification.

recurrent neural networks to elicit the basic features of interest, which uses stacked fully connected neural networks to achieve classification of network traffic. Ren et al. [20] proposed a tree-structured recurrent neural network that uses a tree structure to divide large classification into small classification problems. The model can automatically learn the nonlinear relationship between the input data and the output data, which has a better classification effect. Tal et al. [21] proposed a new method for encrypted traffic classification. The method first converts traffic data into intuitive images, and then combines convolutional neural networks to achieve classification of the images to achieve traffic classification. Li et al. [22] proposed a bidirectional independent recurrent neural network with parallel operations and adjustable gradients to solve the problem that recurrent neural networks are prone to gradient explosion or disappearance. The model extracts the bi-directional structural features of network traffic by forward and backward inputs and combines global attention to emphasize the important features of network traffic. Lin et al. [23] proposed a multi-level feature fusion model to deal with the data imbalance problem. The model combines data timing, byte and statistical features for higher performance. Lin et al. [24] proposed a traffic classification model TSCRNN based on spatial and temporal features. The model first preprocesses the original data, and then learns the spatial and temporal features of the traffic by CNN and bi-directional RNN respectively to achieve efficient classification of the traffic. Saadat et al. [25] proposed a deep learning integrated model. The model first uses a one-dimensional convolutional neural network to automatically extract traffic features, which is then combined with ALO for efficient feature selection and SOM-based clustering to achieve classification of network traffic.

The above models are mostly the integrated use of deep neural networks, convolutional neural networks and recurrent neural networks, which extend the research of deep learning on the field of network traffic classification. However, there is not much research on feature similarity, and the proposed model has a single structure, which cannot extract features in multiple dimensions and fields of view. Also multiple pooling of convolutional neural networks reduces the correlation of network traffic. The ABS-CNN proposed in this paper covers these shortcomings.

III. PROPOSED METHODOLOGIES

The proposed abnormal traffic detection model ABS-CNN is divided into four parts, which are the input part of network traffic, the data preprocessing part, the big-step convolutional model for extracting traffic features and the convolutional attention mechanism for enhancing features. As shown in Figure 1.

A. ABNORMAL TRAFFIC DETECTION MODEL

An input layer, three convolutional layers, a fully connected layer and an output layer are set in the ABS-CNN model, and a convolutional attention mechanism is introduced to enhance the ability of convolution to extract traffic features. The raw traffic is firstly sliced and cleaned to generate a single-channel grayscale image of dimension 28×28. The image data are then histogram equalized and superposed with the original image to generate a $28 \times 28 \times 2$ multichannel grayscale image. The images are then input to the abnormal

traffic detection model. The input layer is responsible for receiving the image data from the network traffic and passing the data to the back layer network, which receives the image data with a dimension of $28 \times 28 \times 2$. The first convolutional layer is set with 64 convolutional kernels of size 7×7 and the step size is set to 2 in the convolutional operation. After the image data passes through this convolution layer, its dimensional size becomes $14 \times 14 \times 64$. The second convolutional layer is set to 128 convolutional kernels of size 5×5, and the step size is still set to 2. After the operation, the data dimension becomes $7 \times 7 \times 128$. Considering the principle that the size of convolutional kernels cannot exceed the feature map, 256 convolutional kernels of size 3×3 are set in the last convolutional layer. With a fully connected layer containing 1600 neurons, all nodes are all connected. Features are classified and summarized at the fully connected layer. Finally, 10 neuron nodes are set in the output layer, in which a sigmoid function is used to output the classification results. The attention mechanism is applied after the first convolutional layer and after the last convolutional layer to enhance the features of the network traffic. To prevent the gradient explosion problem, the ReLU functions are chosen for the activation functions of the three convolutional layers. A cross-entropy error loss function is used as a measure of error in the model training process. Considering the huge number of samples, many data categories, and very complex relationships between features. Therefore, the Adam optimization method, which can adaptively adjust the learning rate, is chosen to learn the parameters of each layer in the model.

Based on the variation of feature map size, it is easy to see that big-step convolution rapidly reduces the feature map dimension.

B. DATA PREPROCESSING

Data preprocessing is an important part of ensuring the quality of the experimental input. The datasets used for the experiments in this paper are USTC-TFC2016 and the real dataset (see Section IV for details), combined with the tool set ''USTC-TK2016'' for data pre-processing. The specific process is shown in Figure 2.

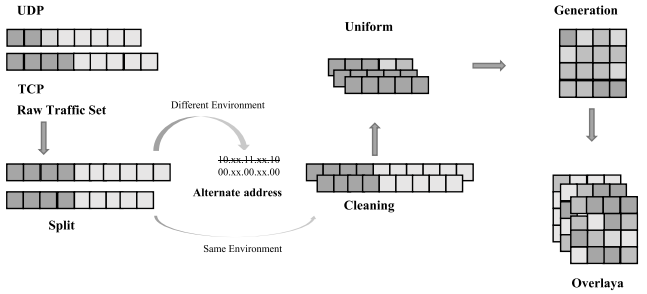

FIGURE 2. Data preprocessing flow chart.

Step 1 (Slice traffic): The raw flow input format is ''pcap'', which is cut into smaller pieces of data by selecting ''All + Flow''. To determine the transport layer protocol, the Transport Control Layer Protocol (TCP) header is 20 bytes, and the User Datagram Protocol (UDP) is 8 bytes. It is necessary to make the TCP and UDP segment headers equal by appending zeros to the UDP header. The file output format is ''pcap''.

Step 2 (Clean up the flow): Eliminate certain information that can affect classification, such as MAC addresses of the data link layer and IP addresses of the IP protocol layer. And replace the original address with a random new address. If the traffic comes from the same environment, there is no need to replace it. Delete also bin files without application layer data and delete flows with the same content.

Step 3 (Generate images): The cleaned file is intercepted as N bytes (n is the grayscale image edge length, $N = n^2$). If the file length is greater than N, it will be intercepted, and less than N needs to be followed by a complementary 0 operation. The uniform length file is constructed as a two-dimensional grayscale image in binary format.

Step 4 (Image overlay): Attentional mechanisms require multi-channel attentional weighting. And step 3 generates grayscale images with the same values for all three channels. Therefore, the histogram equalization enhancement is used to obtain a new image with uniform gray scale, and then superimposed with the original image of one channel to become a multi-channel gray scale image, which is convenient for the channel attention operation.

C. CONVOLUTIONAL ATTENTION MODULE

To improve the model's focus on the traffic features, the traffic grayscale image F is applied to the Convolutional Block Attention Module (CBAM) to obtain a distinctive grayscale image F'', so as to perform fast and accurate classification by ABS-CNN.

CBAM [26], namely the convolutional attention module, which can be incorporated into any CNN to improve the performance of this model. The key to CBAM is to guide the CNN model to focus on important features and suppress nonessential features, which consists of two complementary modules, channel attention and spatial attention. The convolution operation also requires the combined extraction of features from both modules. Figure 3 shows its architecture. The sequential use of both the channel and space modules shows the importance of the intermediate features of the two dimensions.

During the convolution operation, any channel of intermediate features is considered as a feature detector, and the relationship between channels is used to decide which channel is more meaningful. Channel attention [27]assigns more weight to channels with more or distinct features, while assigning the least weight to channels with the least or insignificant features. Unlike channel attention, spatial attention uses the spatial relationships of the original features to help the network localize to the locations of the generated fine features.

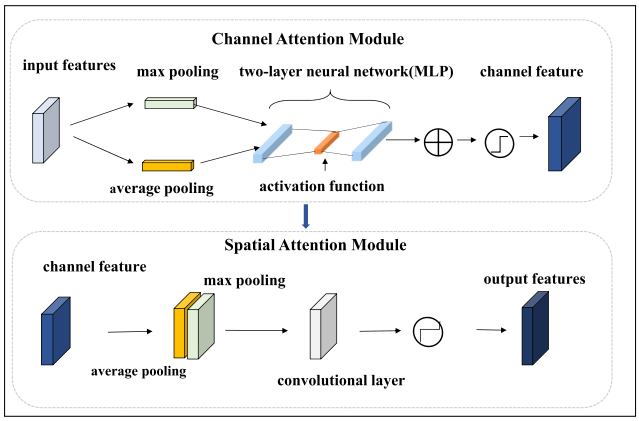

FIGURE 3. CBAM architecture diagram.

The image $F \in R^{C\times H\times W}$ is processed under the channel attention to obtain the channel attention map $M_c \in R^{C\times 1\times 1}$. M_c performs attention weighting on each channel of image F to obtain the feature-refined image F'. The spatial attention map $M_s \in R^{1\times H\times W}$ is then obtained through the processing of the spatial attention module. Then the images F' and $M_s(F')$ are multiplied to obtain F''. Expressed as follows:

$$F' = M_C(F) \otimes F \tag{1}$$
$$F'' = M_s(F') \otimes F' \tag{2}$$

$\otimes$ represents the multiplication of matrices. C represents the number of input feature channels. W represents the width of the image. H represents the height of the image.

For the channel attention module, we first apply average pooling on the image (express withP_{avg})and max pooling (express with P_{max}) to compress the spatial dimension of the intermediate features. Two different features F^c_{max} and F^c_{avg} are collected separately. Next, the two types of features are transformed using a shared multi-layer perceptron (MLP, denoted by P_{MLP}), and the elements of the two features are added. The channel attention map $M_c \in R^{C\times 1\times 1}$ is then obtained by fusing the features with a sigmoid activation function. The formula is as follows:

$$\begin{aligned} M_C(F) &= sigmoid(P_{MLP}(P_{avg}(F)) + P_{MLP}(P_{max}(F))) \\ &= sigmoid(W_1(W_0(F^c_{avg})) + W_1(W_0(F^c_{max}))) \end{aligned} \tag{3}$$

W_0 and W_1 represent the parameters of the two layers of the multilayer perceptron respectively, and the features between the two layers are processed by the activation function ReLU. Finally, $M_c(F)$ is multiplied with its input features to obtain the fine feature image F' after channel attention processing.

The spatial attention module performs average pooling to obtain feature map $F^s_{avg} \in R^{1\times H\times W}$ and maximum pooling to obtain feature map $F^s_{max} \in R^{1\times H\times W}$ for the fine-grained feature image F' processed by the channel attention module. The two feature maps are then concatenated to produce a new feature map. The spatial attention map M_s is obtained after 7×7 convolution kernel and sigmoid activation function operation transformation. The formula is as follows:

$$\begin{aligned} M_s(F) &= sigmoid(f^{7\times 7}([(P_{avg}(F); P_{max}(F)])) \\ &= sigmoid(f^{7\times 7}([F^s_{avg}; F^s_{max}])) \end{aligned} \tag{4}$$

$f^{7\times 7}$ represents the convolution operation with a 7×7 convolution kernel. Finally, the final feature map F'' is generated by multiplying $M_s(F')$ with its input features.

D. BIG STEP CONVOLUTIONAL NEURAL NETWORK

Convolutional Neural Networks use convolution kernels to perform convolution operations on images to extract local features. The traditional convolutional neural network is mainly composed of three parts: convolutional layer, pooling layer, and fully connected layer. Convolution and pooling are two structures unique to convolutional neural networks. Compared with the traditional neural network, the convolutional neural network has three improvements: local receptive field, weight sharing, and pooling layer.

Local receptive field focuses on the local features of the image, and multiple neurons are connected in high layers. The network traffic features are relatively distinct after attention weighting and can be well classified by local features. Weight sharing is performing convolution operations on images with the same convolution kernel, which can reduce the training parameters of the network, making the network structure simpler and more adaptable.

Considering that multiple pooling can cause information loss, resulting in a decrease in classification accuracy. One layer of convolution leads to a small amount of extracted feature data, and too many layers of convolution can lead to overfitting of the model. The amount of experimental data is huge, and the step length is too short leading to a slow training speed. In summary, a big convolutional ABS-CNN with three convolutional layers is designed in this paper.

1) CONVOLUTIONAL LAYER

The convolutional layer is the core constituent structure in a convolutional neural network. The layer integrates a few convolution cores, which contains many neurons. Each neuron is connected to a small region of the upper network, the size of which is the size of the convolutional kernel, also known as the receptive field. The connection between layers in the network is a local connection. The parameters of the convolution kernel also include the size of the convolution kernel, channels and number of channels. The common convolution kernel size is 3×3 or 5×5. The number of channels of the convolution kernel is equal to the number of channels of its input image. For example, the input of ABS-CNN is a multi-channel superimposed grayscale image, so the number of channels is 2. The convolutional layer can deeply analyze the local parts of the neural network to obtain more abstract features. The manifestations are:

$$X^l_j = F(\sum\nolimits_{i\in M_j} X^{l-1}_j K^l_{ij} + b^l_j) \tag{5}$$

l is the current layer. k is the convolution kernel. b is the bias layer for the current layer. M_j is the convolution window of the corresponding jth convolution kernel. The activation function is usually sigmoid, tanh or ReLU.

2) FULLY CONNECTED LAYER

The fully connected layer usually trains a classifier. The previously learned features are input to the classifier and the result of classification is output. The fully connected layer is generally located at the end of the convolutional neural network.

ABS-CNN consists of three layers of convolution and one layer of full connectivity. Table 1 shows the parameters of the three convolutional layers of the convolutional neural network in this paper.

TABLE 1. CNN parameters.

layer	convolution kernel	step size	activation	padding
1	7×7	2	ReLU	same
2	5×5	2	ReLU	same
3	3×3	1	ReLU	same

IV. EXPERIMENT

A. DATA SET AND EXPERIMENTAL ENVIRONMENT

1) EXPERIMENTAL DATASET

This paper requires the raw traffic to be used which includes both normal and malicious traffic for the experimental study. Therefore, the public dataset USTC-TFC2016 created by Wang et al. [28] is used as the experimental dataset in this paper. The dataset consists of two parts, one for the 10 malicious flows in CTU-13 and the other for the 10 normal flows collected by IXIS BPS devices. The dataset covers a wide range of common networks and applications, reflecting the diversity of the data. The sample balance gives the model good generalization ability. Therefore, we select 10 kinds of traffic data, of which 5 kinds of normal traffic and 5 kinds of malicious traffic, 20,000 pieces of each traffic in this paper. Details are shown in Table 2 and Table 3.

TABLE 2. USTC-TFC2016 malicious traffic statistics.

Name	CTU number	NetFlow quantity	File size	Handle method
Geodo	119-2	25000	28.8 MB	Original file interception
Miuref	127-1	88560	16.4 MB	Original file
Shifu	142-1	500000	57.9 MB	Original file interception
Tinba	150-1	22000	2.55 MB	Original file interception
Zeus	116-2	25000	28.8 MB	Original file

2) REAL DATASET

The real dataset also contains both normal and malicious traffic. Normal traffic is captured by Microsoft Network Monitor for 5 kinds of application software for daily use. There are totally 5,000 samples and 1,000 samples in each category including web page (Baidu), communication (QQ and QQ email), video (IQIYI) and social (Weibo). Hping3 tool is used to simulate DDoS attacks in the experiments and Wireshark is used to collect DDoS traffic in the attack state. The ARP attack simulation experiment [29] is then performed, also using Wireshark to collect ARP traffic. Together with the other three types of malicious traffic of CTU-13, the number of malicious traffic samples is 5000.

TABLE 3. USTC-TFC2016 normal traffic statistics.

Name	NetFlow quantity	File size	Type
BitTorrent	15000	7.33MB	P2P
WorldOfWarcraft	140000	14.9MB	Electronic games
Gmail	25000	9.05MB	E-mail
Outlook	15000	11.1MB	E-mail
Skype	12000	4.23MB	Instant messaging

The data information of malicious traffic (see Table 4) and normal traffic (see Table 5) collected by Microsoft Network Monitor is as follows.

In the experiments of this paper, the data set is divided into training set, validation set and test set in the ratio of 7:1:2. And based on the performance of the model in the validation set, we evaluate the fitting ability of the model and select the best model. The test set is used to evaluate the generalization ability of the final model.

TABLE 4. Actual malicious traffic statistics.

Name	NetFlow quantity	File size	Type
DDos	41548	4.7 MB	Original file interception
ARP	49218	6.1 MB	Original file interception
Htbot	171569	83.6 MB	Original file
Nsis-ay	352266	281 MB	Original file
Virut	440625	109 MB	Original file

TABLE 5. Actual normal traffic statistics.

Name	NetFlow quantity	File size	Type
Baidu	6000	2.40 MB	multimedia streaming
IQIYI	360000	60.2 MB	media video
QQ	771887	1 GB	Instant messaging
QQemail	25000	9.05 MB	E-mail
Weibo	756068	1 GB	Social network

3) EXPERIMENTAL ENVIRONMENT

The ABS-CNN model proposed in this study is modeled based on the current mainstream deep learning framework TensorFlow [30] and python3.6 programming language in Windows 10 operating system. Training of ABS-CNN model is implemented in a computer with i5-8300H CPU, GTX1050 Ti GPU with 8GB RAM.

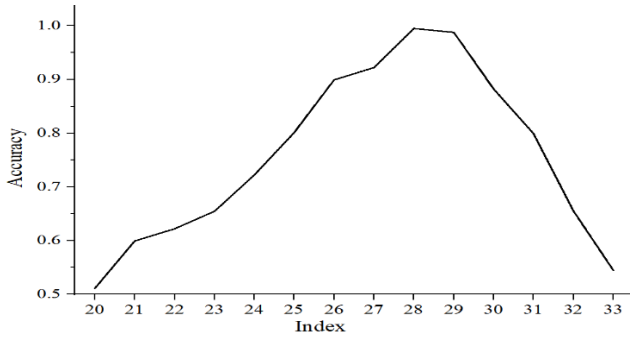

FIGURE 4. Classification accuracy corresponding to different n values.

B. BYTE LENGTH SELECTION AND MODEL SETTING

1) MODEL PARAMETER SETTING

In this paper, several different combinations of parameters were tried for the experiment. The results show that the ABS-CNN model has the highest accuracy when the epoch is set to 20, the batch size is set to 20, and the number of convolutional layers is 3. The value of the loss function of the model decreases and eventually stabilizes with training, and no local minimum occurs during the training process, which indicates the reliable performance of the model.

2) TRAFFIC BYTE LENGTH AND ANALYSIS

The length chosen for the traffic data is critical to the experimental results. Using full-length traffic data increases the calculation overhead of the algorithm, and traffic data is too short which reduces the accuracy of the algorithm. Analyzing the composition structure of the whole data set, the shortest length is 412 bytes and the longest is 1058 bytes, so the value range of n is set to [20, 33]. Experiments were conducted to observe the influence of different n values on classification results, so as to determine the best value of N (the best length of traffic data). In this experiment, accuracy is used as the performance metric, and the experimental results are shown in Figure 4.

Figure 4 represents the accuracy of the model classification for different values of n. From the figure, it is easy to know that the model classification accuracy is highest when n takes the value of 28, and the traffic data length is 784 (N is 784).

C. EXPERIMENTAL PERFORMANCE EVALUATION INDEX

The evaluation index of the experimental results contains two parts: (1) the accuracy rate, the recall rate, and the F1-score; (2) the accuracy rate of the model.

$$\text{Accuracy of class iprecision} = \frac{TP}{TP + FP} \tag{6}$$

$$\text{Recall of class } i \text{ recall} = \frac{TP}{TP + FN} \tag{7}$$

$$\text{F1-Score } F_1 = \frac{2 \times precision \times recall}{precision + recall} \tag{8}$$

$$\text{Model Accuracy } A = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

TABLE 6. Comparative analysis of ablation experiments.

Model	Accuracy	Precision	Recall	F1-Score
NA-ABS-CNN	94.39%	92.15%	94.21%	94.1%
NH-ABS-CNN	94.17%	91.37%	92.13%	92%
WP-ABS-CNN	93.13%	92.41%	93%	92.75%
ABS-CNN	**99.5%**	**99.37%**	**99.71%**	**99.53%**

D. PERFORMANCE COMPARISON

1) PERFORMANCE COMPARISON OF DIFFERENT COMPONENT MODELS

In this paper, the ablation study is performed by removing each component in turn from the proposed ABS-CNN and comparing it with the ABS-CNN of the complete pair to verify the impact of each component on the model. To examine the effects of attention mechanism, histogram equalization, and large-step convolution on model performance. This paper sets up three ablation models: ABS-CNN without attention mechanism (NA-ABS-CNN), ABS-CNN without histogram equalization (NH-ABS-CNN), and ABS-CNN with pooling (WP-ABS-CNN). To visualize the effect of each component on the model, the ablation study is performed on the experimental dataset and the model is trained and validated based on the same training and test sets. The experimental results are shown in Table 6, with the best performance in bold font.

As can be seen from Table 6, ABS-CNN maintains the maximum value in terms of accuracy, precision, recall and F1-Score parameters. The accuracy of ABS-CNN is 99.5%, which is 5.11% higher compared to the accuracy of NA-ABS-CNN. It is shown that the attention mechanism assigns different weights to similar features, enhances the differentiation of features, improves the detection performance of the model, and provides research feasibility for the traffic classification problem with similar features. The accuracy of ABS-CNN is 5.33% higher than that of NH-ABS-CNN, which proves that histogram equalization improves the superiority of model detection. Histogram equalization enhances the image and overlays it with the original image to turn the single-channel image into a multi-channel image, which enhances the features of the traffic and makes up for the lack of a single dimension of the detection model. Compared to WP-ABS-CNN, the accuracy of ABS-CNN increased by 6.37%. Network traffic is sequence correlated data, multiple pooling can reduce the correlation of traffic data, and removing the pooling layer can retain the complete traffic information. Multi-channel convolution can effectively and comprehensively extract the higher-order features of the traffic, which is beneficial to the detection performance of ABS-CNN.

2) PERFORMANCE COMPARISON OF DIFFERENT MODELS AND ROBUSTNESS TESTING OF THE MODELS

The following experiments were designed to verify the anomalous traffic detection performance of ABS-CNN. The experiments were performed on the experimental data set.

- Experiment 1: To verify the superior performance of the ABS-CNN model, the training dataset is applied to other current more advanced machine learning models: support vector machine (SVM), artificial neural network (ANN), random forest (RF), Naive Bayes (Bayes) and convolutional neural network (CNN). And validate all models in the same test set. In this study, the accuracy of SVM in the test set is calculated and used as a baseline to evaluate the performance of other machine learning models.
- Experiment 2: In order to evaluate the performance of ABS-CNN model more intuitively and clearly, two models with high correlation degree are selected for comparison in this paper. (1) APSO-CNN [31] utilizes the PSO algorithm that varies with inertia for adaptive optimization of the structural parameters of a one-dimensional CNN. (2) PBCNN [32] automatically extracts abstract features at bytes in packets and performs further multi-classification for abnormal traffic detection between sessions and packets in streams.
- Experiment 3: In order to verify the efficiency of ABS-CNN in the field of encrypted malicious traffic detection and classification, this paper conducts experiments based on real dataset. The real dataset contains five types of normal traffic captured by real environments, two types of malicious traffic captured by simulated attacks and three types of malicious traffic from CTU-13.

3) PERFORMANCE COMPARISON RESULTS AND ANALYSIS

After performing Experiment 1, the performance results of the proposed model on each parameter are shown in Table 7. Because the computation times of ML and DL algorithms are not comparable, no runtime analysis was performed for Experiment 1.

TABLE 7. Performance comparison of abs-cnn with multiple models.

Model	Accuracy	Precision	Recall	F1-Score
SVM	80.02%	77.21%	83.91%	80.21%
Bayes	92.92%	92.01%	92.6%	92%
RF	93.35%	93.33%	93.21%	93.3%
ANN	91.62%	92.12%	90.05%	91.11%
CNN	81.88%	83.71%	81.59%	82%
ABS-CNN	**99.5%**	**99.37%**	**99.71%**	**99.53%**

As can be seen from Table 7, SVM has the lowest accuracy rate. However, its recall is not the lowest, so it is difficult to evaluate the performance of the model based on accuracy and recall alone. The F1-Score is the weighted harmonic mean of accuracy and recall, so we use the F1-Score and accuracy combined to evaluate the performance of the model.

SVM has the lowest F1 and its performance is similarly the worst. The F1 scores of the remaining four models are all below 94%, which makes it difficult to provide proper guidance for abnormal traffic detection. In comparison,

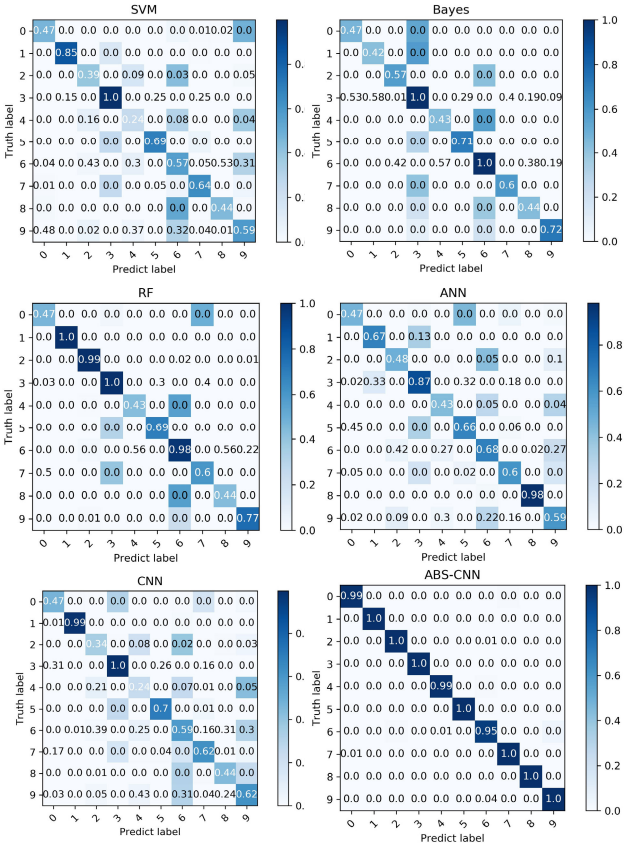

FIGURE 5. Confusion matrix for different models.

ABS-CNN maintains a maximum accuracy, precision, recall and F1-Score of over 99%. Compared with other models, the performance of ABS-CNN has been improved. This is because SVM, Bayes, RF, ANN are shallow machine learning algorithms with weak techniques for deep feature mining. The ordinary convolutional neural network cannot effectively extract the higher-order features of the traffic for traffic with similar features. And pooling reduces the sequential correlation of traffic, so ordinary CNNs do not perform well in classifying traffic with similar features. ABS-CNN can effectively learn higher-level feature representations in sequential data and has stronger nonlinear fitting ability, which is suitable for accurate classification of network traffic under high-dimensional data. At the same time, the multi-dimensional features of network traffic can be effectively and comprehensively extracted by stacking multiple layers of convolution. And the attention mechanism is introduced to assign different weights to similar features, and the histogram equalization enhances the dimensionality of the features, which improves the performance of the whole model.

For a more intuitive view of the classification accuracy of ABS-CNN models for various types of traffic, Figure 5 shows the confusion matrix for different models.

As can be seen from Figure 5, the ABS-CNN model performs very well in classifying the remaining nine types of traffic, except for the Shifu type, which are all above 99%, outperforming the other models. And the classification accuracy of Shifu is 95%, which is due to the confusion of Shifu traffic. Four percent of Shifu was incorrectly identified as Geodo traffic, and one percent was identified as Zeus traffic.

On the other models, the same small percentage of Shifu traffic is identified as Geodo and Zeus traffic, which may be related to the application characteristics of Shifu. Bayes performs best on the classification of Shifu traffic, but does not perform well on the classification of other traffic, so it is less useful. Among the other 9 types of traffic, except for BitTorrent and Miuref, the classification accuracy of the remaining 7 types of traffic is 100%, which achieves accurate classification. BitTorrent and Miuref also achieved a classification accuracy rate of 99%, with only 1% of traffic being misclassified. Overall, ABS-CNN performs far better than other models and achieves the best classification results.

In Experiment 2, each model completes 20 epochs in the same training set with the batch size set to 20 as well, which is tested based on the same test set. The experimental results are shown in Table 8.

TABLE 8. Comparison of operational efficiency of different models.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Training time (s)	Test time(s)
APSO-CNN	93.95	91.72	94.1	93.21	**943**	**1.64**
PBCNN	94.06	93.01	94.51	94	1092	2.7
ABS-CNN	**99.5**	**99.37**	**99.71**	**99.53**	950	1.69

As can be seen from Table 8, the training time of the ABS-CNN proposed in this paper is 950s, which is 142s less than that of the PBSNN. It is almost identical to APSO-CNN, which has the simplest model structure, and only has 7s more than APSO-CNN. In terms of test time, ABS-CNN is 1.01s faster than PBSNN and 0.05s slower than APSO-CNN. Compared with PBSNN, ABS-CNN has faster training and testing speed because removing the pooling layer reduces the training parameters. APSO-CNN has the shortest training time and testing time because of its one-dimensional convolution and simple structure. The structure of ABS-CNN is relatively more complex than that of APSO-CNN, but the training time and testing time are similar to APSO-CNN, which proves that ABS-CNN runs fast. In terms of accuracy, ABS-CNN is 5.55 % higher than APSO-CNN and 5.44% higher than PBCNN. And ABS-CNN outperforms other models in terms of accuracy, precision, recall and F1-Score, which are all above 99%. Demonstrate the high sensitivity of ABS-CNN for abnormal traffic detection. And ABS-CNN has higher operational efficiency while ensuring classification accuracy.

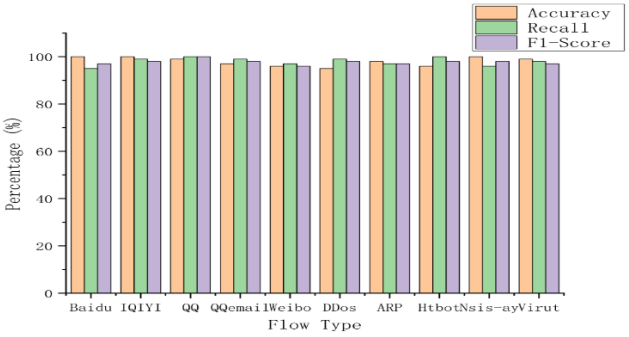

FIGURE 6. Traffic classification results based on actual dataset.

In Experiment 3, the real environment captured traffic is similarly preprocessed and then input to the model for validation. The classification results for each type of flow are shown in Figure 6.

The classification accuracy of ABS-CNN on the real dataset is 97.65% with excellent classification results. As shown in Figure 6, the accuracy, recall and F1-Score of each flow rate reached over 97%. Among them, the accuracy of Baidu, IQIYI and abnormal traffic Nsis-ay is 100%, which achieves perfect classification. The FI-Score of the other 7 types of traffic are all above 97%, which lays the foundation for future classification of real-time traffic. And the traffic acquired by the real environment is encrypted traffic. ABS-CNN not only enables efficient classification of encrypted traffic, but also has the ability of fine-grained classification of encrypted malicious traffic. Therefore, it can be said that ABS-CNN can be applied to different complex environments and has a certain degree of robustness.

V. CONCLUSION

To address the difficulties caused by similar features and single model structure on abnormal traffic detection, this paper proposes a detection model based on attention and big-step convolution. Experiments were conducted on both publicly available dataset and real environment crawls dataset. The efficiency of the model is seen through performance analysis.

- ABS-CNN is the highest in accuracy, precision, recall and F1-Score when compared with traditional models. It is proved that ABS-CNN achieves high accuracy and prediction with good detection effect. And from the confusion matrix of various types of traffic, the classification accuracy of multiple traffic is 100%, which reflects the high sensitivity of ABS-CNN in abnormal traffic detection.
- Compared with different variants of CNN models, ABS-CNN has a shorter training time as well as testing time and runs efficiently. And ABS-CNN shows unparalleled advantages in accuracy, precision, recall and F1-Score with the best classification results.
- The results of the ablation analysis show that ABS-CNN introduces an attention mechanism to assign attention weights for different features, which

enhances the differentiation of features and relieves the difficulties caused by feature similarity. ABS-CNN introduces histogram equalization in data preprocessing, which improves the structure of single channel in the model and enhances the detection performance of the model. At the same time, removing the pooling layer retains the sequence-related features, which reduces the training parameters, improves the operation efficiency and achieves efficient abnormal traffic detection.

- ABS-CNN experiments on traffic crawled by real environment and has excellent detection results. The traffic captured by the real environment is encrypted traffic. ABS-CNN not only achieves efficient classification of encrypted traffic application types, but also reflects the fine-grained ability to encrypt malicious traffic. This demonstrates that the ABS-CNN is able to adapt to environments of different complexity and has a degree of robustness.

The proposed algorithm deepens the application of attention mechanism and histogram equalization on abnormal traffic detection, and also proposes a possible solution for the difficulties of similar features and single model dimension on abnormal traffic detection. The following are future research directions:

- Data pre-processing still requires splitting by current network tools to obtain samples, which results in a small number of samples lost. In addition, the five-tuple sequence resulted in duplicate and unlabeled invalid samples. Do further research in the future to find more suitable pre-processing sequences and tools.
- Analyze the temporal and spatial relationships of different packets to study anomalous traffic detection in spatial and temporal mining.

REFERENCES

[1] O. Salman, I. H. Elhajj, A. Kayssi, and A. Chehab, "A review on machine learning–based approaches for internet traffic classification," *Ann. Telecommun.*, vol. 75, nos. 11–12, pp. 673–710, Dec. 2020.

[2] A. Madhukar and C. Williamson, "A longitudinal study of P2P traffic classification," in *Proc. 14th IEEE Int. Symp. Modeling, Anal., Simulation*, Monterey, CA, USA, Sep. 2006, pp. 179–188, doi: 10.1109/MASCOTS.2006.6.

[3] S. Sen, O. Spatscheck, and D. Wang, "Accurate, scalable in-network identification of P2P P2P traffic using application signatures," in *Proc. 13th Int. Conf. World Wide Web*, New York, MY, USA, May 2004, pp. 512–521.

[4] L. Ding, J. Liu, T. Qin, and H. Li, "Internet traffic classification based on expanding vector of flow," *Comput. Netw.*, vol. 129, pp. 178–192, Dec. 2017.

[5] T. Liu, Y. Sun, and L. Guo, "Fast and memory-efficient traffic classification with deep packet inspection in CMP architecture," in *Proc. IEEE 5th Int. Conf. Netw., Archit., Storage*, Macau, China, Jul. 2010, pp. 208–217, doi: 10.1109/NAS.2010.43.

[6] N. Cascarano, L. Ciminiera, and F. Risso, "Optimizing deep packet inspection for high-speed traffic analysis," *J. Netw. Syst. Manage.*, vol. 19, no. 1, pp. 7–31, Mar. 2011.

[7] G. Aceto, A. Dainotti, W. de Donato, and A. Pescape, "PortLoad: Taking the best of two worlds in traffic classification," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM)*, San Diego, CA, USA, Mar. 2010, pp. 1–5, doi: 10.1109/INFCOMW.2010.5466645.

[8] L. Vu, C. T. Bui, and Q. U. Nguyen, "A deep learning based method for handling imbalanced problem in network traffic classification," in *Proc. 8th Int. Symp. Inf. Commun. Technol.*, Dec. 2017, pp. 333–339.

[9] P. Wang, F. Ye, X. Chen, and Y. Qian, "Datanet: Deep learning based encrypted network traffic classification in SDN home gateway," *IEEE Access*, vol. 6, pp. 55380–55391, 2018.

[10] J. H. Shu, J. Jiang, and J. X. Sun, "Network traffic classification based on deep learning," *J. Phys., Conf. Ser.*, vol. 1087, Sep. 2018, Art. no. 062021.

[11] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.

[12] C. Wang, T. Xu, and X. Qin, "Network traffic classification with improved random forest," in *Proc. 11th Int. Conf. Comput. Intell. Secur. (CIS)*, Shenzhen, China, Dec. 2015, pp. 78–81, doi: 10.1109/CIS.2015.27.

[13] Z. Yuan and C. Wang, "An improved network traffic classification algorithm based on Hadoop decision tree," in *Proc. IEEE Int. Conf. Online Anal. Comput. Sci. (ICOACS)*, Chongqing, China, May 2016, pp. 53–56, doi: 10.1109/ICOACS.2016.7563047.

[14] A. V. Phan, M. L. Nguyen, and L. T. Bui, "Feature weighting and SVM parameters optimization based on genetic algorithms for classification problems," *Appl. Intell.*, vol. 46, no. 2, pp. 455–469, Mar. 2017.

[15] B. Schmidt, A. Al-Fuqaha, A. Gupta, and D. Kountanis, "Optimizing an artificial immune system algorithm in support of flow-based internet traffic classification," *Appl. Soft Comput.*, vol. 54, pp. 1–22, May 2017.

[16] S. Dong, "Multi class SVM algorithm with active learning for network traffic classification," *Expert Syst. Appl.*, vol. 176, Aug. 2021, Art. no. 114885.

[17] J. Cao, Z. Fang, G. Qu, H. Sun, and D. Zhang, "An accurate traffic classification model based on support vector machines," *Int. J. Netw. Manage.*, vol. 27, no. 1, Jan. 2017, Art. no. e1962.

[18] D. Md. Farid, N. Harbi, and M. Zahidur Rahman, "Combining Naive Bayes and decision tree for adaptive intrusion detection," 2010, *arXiv:1005.4496*.

[19] G. D'Angelo and F. Palmieri, "Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction," *J. Netw. Comput. Appl.*, vol. 173, Jan. 2021, Art. no. 102890.

[20] X. Ren, H. Gu, and W. Wei, "Tree-RNN: Tree structural recurrent neural network for network traffic classification," *Expert Syst. Appl.*, vol. 167, Apr. 2021, Art. no. 114363.

[21] T. Shapira and Y. Shavitt, "FlowPic: A generic representation for encrypted traffic classification and applications identification," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 2, pp. 1218–1232, Jun. 2021.

[22] H. Li, H. Ge, H. Yang, J. Yan, and Y. Sang, "An abnormal traffic detection model combined BiIndRNN with global attention," *IEEE Access*, vol. 10, pp. 30899–30912, 2022.

[23] K. Lin, X. Xu, and F. Xiao, "MFFusion: A multi-level features fusion model for malicious traffic detection based on deep learning," *Comput. Netw.*, vol. 202, Jan. 2022, Art. no. 108658.

[24] K. Lin, X. Xu, and H. Gao, "TSCRNN: A novel classification scheme of encrypted traffic based on flow spatiotemporal features for efficient management of IIoT," *Comput. Netw.*, vol. 190, May 2021, Art. no. 107974.

[25] S. Izadi, M. Ahmadi, and R. Nikbazm, "Network traffic classification using convolutional neural network and ant-lion optimization," *Comput. Electr. Eng.*, vol. 101, Jul. 2022, Art. no. 108024.

[26] Y. Wang, Z. Zhang, L. Feng, Y. Ma, and Q. Du, "A new attention-based CNN approach for crop mapping using time series Sentinel-2 images," *Comput. Electron. Agricult.*, vol. 184, May 2021, Art. no. 106090.

[27] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.

[28] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," in *Proc. IEEE Int. Conf. Intell. Secur. Informat. (ISI)*, Beijing, China, Jul. 2017, pp. 43–48, doi: 10.1109/ISI.2017.8004872.

[29] N. Ahuja, G. Singal, D. Mukhopadhyay, and A. Nehra, "Ascertain the efficient machine learning approach to detect different ARP attacks," *Comput. Electr. Eng.*, vol. 99, Apr. 2022, Art. no. 107757.

[30] M. Abadi, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*.

[31] X. Kan, Y. Fan, Z. Fang, L. Cao, N. N. Xiong, D. Yang, and X. Li, "A novel IoT network intrusion detection approach based on adaptive particle swarm optimization convolutional neural network," *Inf. Sci.*, vol. 568, pp. 147–162, Aug. 2021.

[32] L. Yu, J. Dong, L. Chen, M. Li, B. Xu, Z. Li, L. Qiao, L. Liu, B. Zhao, and C. Zhang, "PBCNN: Packet bytes-based convolutional neural network for network intrusion detection," *Comput. Netw.*, vol. 194, Jul. 2021, Art. no. 108117.

DAOQUAN LI received the B.S. degree in semiconductor physics and devices from the Shandong University of Industry, in 1991, the M.S. degree in microsystem electronics from Xian Jiaotong University, in 1994, and the Ph.D. degree in computer software and theory from the Shandong University of Science and Technology, in 2011.

From 1994 to 2005, he was with CITIC Securities Company Ltd., (Shandong). Since 2005, he has been with the School of Information and Control Engineering, Qingdao University of Technology. He was a professor, in 2011. He has published more than 50 articles in wireless sensor networks, ad-hoc networks, software define networks, edge computing, and e-commerce.

Prof. Li is a member of CCF and Communication Society, an Expert Member of ACM Jinan Branch, a China Senior Registered Information Security Engineer, and an Evaluation Expert of China Postdoctoral Science Foundation.

XUEQING DONG was born in Hebei, China, in 1997. She received the B.S. degree in software engineering from the Qingdao University of Technology, in 2019, where she is currently pursuing the M.S. degree in electronic information. Her current research interests include deep learning and information network security.

JIE GAO was born in Shandong, China, in 1998. She received the bachelor's degree in software engineering from the Qingdao University of Technology, in 2021, where she is currently pursuing the master's degree in electronic information. Her current research interests include deep learning and information network security.

KEYONG HU received the Ph.D. degree in computer science and technology from the Ocean University of China, Qingdao, China, in 2014.

He is currently an Associate Professor with the School of Information and Control Engineering, Qingdao University of Technology, Qingdao. His research interests include sensor networks, machine learning, and big data.