

ENTROPY-BASED FEATURE EXTRACTION ALGORITHM FOR ENCRYPTED AND NON-ENCRYPTED COMPRESSED TRAFFIC CLASSIFICATION

ZHENGZHI TANG^{1,2}, XUEWEN ZENG¹ AND YIQIANG SHENG¹

¹National Network New Media Engineering Research Center
Institute of Acoustics, Chinese Academy of Sciences
No. 21, North 4th Ring Road, Haidian District, Beijing 100190, P. R. China
{ tangzz; zengxw; shengyq }@dsp.ac.cn

²School of Electronic, Electrical and Communication Engineering
University of Chinese Academy of Sciences
No. 19(A), Yuquan Road, Shijingshan District, Beijing 100049, P. R. China

Received July 2018; revised November 2018

ABSTRACT. *The study of network traffic identification is not only important for the network management, but also crucial to monitor network security issues. Currently, traffic classification tasks including protocols identification, applications identification and traffic characterization identification and so on have a good result. However, existing classification methods can hardly distinguish between encrypted and non-encrypted compressed traffic. In this paper, we propose an entropy-based feature extraction algorithm for encrypted and non-encrypted compressed traffic classification, which uses the entropy of fixed-length packet payload. For a fixed-length binary sequence from packet payload, the algorithm uses a sliding window of 8-bit and slides through different bits to obtain different sequences. Then it calculates the serial binary entropy of different sequences and an entropy vector as feature vector of the original sequence is obtained. By this method, the feature vectors of encrypted traffic and non-encrypted compressed traffic sequences are used as input of the support vector machine or random forest for training and classification. The experimental results show that the proposed feature extraction algorithm can well distinguish between encrypted traffic and non-encrypted compressed traffic. When the packet payload length is 1444 bytes, it can reach high classification accuracy (about 97.90%).*

Keywords: Sliding window, Serial binary entropy, Encrypted traffic, Non-encrypted compressed traffic

1. Introduction. Network traffic classification as a means to achieve traffic management appears to be more important. Especially for Internet Service Providers (ISPs), knowing the specific traffic types and proportions transmitted in their networks, they can take appropriate management measures to provide better Quality of Services (QoS) for real-time traffic, and to provide better service for users. Of course, with the growing importance of network security issues, traffic classification is also increasingly important in protocol analyzers for Internet traffic monitoring [1] and intrusion detection systems [2,3]. Currently, traffic classification tasks include protocols identification [4], applications identification [5] and traffic characterization identification [6] and so on.

Although the current researches of traffic identification have achieved good results, existing encrypted traffic identification methods can hardly distinguish between encrypted traffic and non-encrypted compressed traffic. In work of [7], authors use the entropy

computing on consecutive bytes and frequencies of characters as features and a support vector machine algorithm was used to classify traffic. However, classification over partial file space cannot distinguish encrypted files from non-encrypted compressed files. Then an additional heuristic to distinguish these categories using frequencies of four-bit characters was proposed. In work of [8], authors give an advanced improvement classification method for the work of [7]. A Monte Carlo simulation method used to estimate the error of π value was proposed. The error of π value is used as an added feature, which can be used to distinguish the local random traffic and the whole random traffic. However, when the amount of encrypted traffic is small, the identification performance is poor. Moreover, no explicit experimental results about the encrypted and non-encrypted compressed traffic classification were given. In work of [9], authors give a research on the identification of different encrypted and compressed algorithms. However, there is no discussion of the distinction between encrypted and compressed data.

Considering the shortcomings of these methods to distinguish encrypted and non-encrypted compressed traffic, we propose a classification method which uses the entropy of fixed-length packet payload. It calculates the serial binary entropy of different sequences and an entropy vector as feature vector of the original sequence is obtained. By this method, the feature vectors of encrypted traffic and non-encrypted compressed traffic sequences are used as input of the machine learning for training and classification. The experimental results show that the proposed method can well distinguish between encrypted traffic and non-encrypted compressed traffic.

The main contributions of this paper are as follows.

- We propose an entropy-based feature extraction algorithm for encrypted and non-encrypted compressed traffic classification to reduce the case where non-encrypted compressed traffic is misidentified as encrypted traffic and improve the accuracy of encrypted traffic identification.
- To better analyze the characteristics of encrypted and non-encrypted compressed data, we propose a binary sequence sampling and recombination method based on sliding window. For a fixed-length binary sequence from packet payload, it can obtain 8 new binary sequences through sampling and recombination from the original sequence. So, it can mine the characteristics of original data as much as possible.
- We design a method of calculating the consecutive binary subsequence entropy. It is mainly to calculate the entropy value of binary sequence under different scales and combinations. By this method, the entropy feature vector of original sequence is obtained. So, the encrypted and non-encrypted compressed traffic can be distinguished based on the entropy feature vectors.
- We carry out many experiments on the Support Vector Machine (SVM) and Random Forest (RF) to evaluate the performance of the proposed feature extraction algorithm. The sensitivity of this algorithm against different situations is studied. In the best case, the proposal outperforms 4.9% the latest technique in terms of classification accuracy. In the worst case, they are well matched in classification accuracy.

The rest of this paper is organized as follows. Section 2 summarizes the related work. In Section 3, we explain the definition of entropy and the definition of serial binary entropy is proposed in information theory. In Section 4, we give a detailed description of entropy-based feature extraction algorithm proposed and establish an analytical framework based on machine learning to evaluate the feature extraction algorithm performance. In Section 5, we explain the experimental process and analyze the experimental results. Finally, Section 6 concludes the work and analyzes possible future studies.

2. Related Work. When the Internet network technology firstly emerged, traffic in the network was in plain text. Until now, the technology of non-encrypted traffic identification has been very mature. The commercial products, such as Cisco Network Based Application Recognition (NBAR), and Snort, mostly use the payload inspection technique on the traffic payload classification as input to map flows to application protocols [10]. Herein, according to the principle and nature of the methods used in non-encrypted traffic classification, we can categorize these approaches into four main categories as follows. (1) Port-based approach [11]. It extracts the port number which is assumed to be associated with a particular application from the TCP/UDP headers of the packets. (2) Payload inspection approach [12]. It uses rule matching (such as Regular Expressions) or other methods to analyze the application layer payload of packets. (3) Behaviors-based approach [13]. It builds an interaction graphs model from the perspective of application level layer interaction behaviors among hosts and then analyze such interaction graphs with graph theory techniques. (4) Machine learning approach [14-17]. Machine learning approaches, including traditional machine learning and deep learning, are currently the most studied. With the widespread use of GPU and the development of specialized Artificial Intelligence (AI) chips, machine learning methods have become very efficient.

With the increasing awareness of privacy protection, the proportion of encrypted network traffic gradually increases. This change poses a challenge to currently used methods for traffic measurement, for which the identification and analysis of network traffic [10] become more difficult. However, in recent years, there have been some advances in the study of encrypted traffic identification. According to the work of [18], we make an appropriate induction and categorize the encrypted traffic identification methods into five main categories as follows. (1) Payload detection [19,20]. In the work of [19], a new DPI system that can inspect encrypted payload without decryption was proposed, thus solved the user privacy issue, but it can only process HTTP Secure (HTTPS) traffic. In the work of [20], stochastic fingerprints based on first-order homogeneous Markov chains for application traffic flows conveyed in Secure Socket Layer/Transport Layer Security (SSL/TLS) sessions were proposed. (2) Payload randomization and distribution [9,21]. According to the characteristics of the network application traffic is not completely encrypted, and the traffic can be identified by the randomness of the same characteristic fields carried by each packet. (3) Machine learning methods [10,22]. In the work of [10], numerous machine learning methods to identify encrypted traffic were summarized. These machine learning methods do well in identification accuracy. In the work of [22], encrypted traffic was treated as images and sequences, using Convolutional Neural Network (CNN) as classifier and works very well. (4) Behaviors-based approach [23]. In work of [23], a real-time identification method of encrypted P2P traffic based on host behavior association was proposed. However, only a few encrypted applications can actually benefit from this method. (5) Hybrid approach [24]. By combining multiple algorithms, better identification accuracy has been achieved. In work of [24], a method for encrypted traffic identification combined with signature and statistical analysis was proposed, and the experimental results show that the method can identify more than 99% of SSL/TLS traffic.

Table 1 gives a summary of traffic classification methods. It can be seen from the table that different methods are specific to the different traffic characteristics. At the same time, the classification accuracy of each method has certain differences, and the machine learning method has the characteristics of real-time and high accuracy, which has been widely studied.

TABLE 1. The summary of traffic classification methods

Methods	Inspection content	Encryption or not	Classification accuracy (High > Medium > Low)
Port-based	Port number	Non-encryption	Past high, but now low
Payload inspection	Payload	Both	High (Encryption only for HTTPS)
Behaviors	Host behaviors	Both	Medium
Machine learning	Flow statistics features	Both	High
Payload randomization and distribution	Partial payload	Encryption	Medium
Hybrid approach	Many features	Encryption	High

3. Classification Using Entropy Vector. The information entropy method can effectively distinguish the low entropy data from the high entropy data. Therefore, using information entropy is possible to identify high entropy encrypted traffic. However, non-encrypted compressed traffic also has high entropy value, so it is easy to misidentify non-encrypted compressed traffic as encrypted traffic. In this paper, we make use of an entropy vector, which is made up of different dimensions entropy, as classification feature. First of all, definitions of entropy and the serial binary entropy vector which we defined are explained clearly in this part.

3.1. Definitions of entropy. According to the information theory, entropy is a measure of data disorder or data randomness. Assuming that a sequence of n elements is S ($S = \{x_1, x_2, \dots, x_n\}$), the entropy of the sequence S is defined as $-\sum_{i=1}^n p(x_i) \log p(x_i)$, where $p(x_i)$ denotes the frequency of element x_i in set S . In this paper, we use the standardized entropy, defined as $H/\log(n)$, where H denotes the entropy of the sequence S and the normalized factor is the logarithm of n to the base 2. Note that all logarithms in this paper are to the base 2 and for the convenience of calculation, herein $0 \log 0 = 0$ is defined. The minimum entropy value is 0 if all elements in the sequence S are the same, and the maximum entropy value is 1 if all elements are distinct in the sequence S .

3.2. Definitions of serial binary entropy. The randomness of a sequence is measured by overlooking the conditional probability of elements and entropy is calculated by assuming the elements are independent [9]. Hence, for a file or a packet payload, we firstly read fixed-length binary stream as a binary sample sequence. Each element in the binary sample sequence is 0 or 1. Then the binary sample sequence is divided into lots of fixed-length binary subsequences. All the fixed-length binary subsequences form a set S . Each fixed-length binary subsequence is treated as an element x_i in set S , so we can calculate the serial binary entropy of the binary sample sequence. For a more general application, we assume that the length of fixed-length binary subsequence is l , and then we can obtain arbitrary kl ($k = 1, 2, \dots, n$) fixed-length binary subsequence from the given binary sample sequence as an element and calculate the serial binary entropy of given binary sample sequence over all possible k values. In order to better understand the calculation process of serial binary entropy, a simple example is given. We let the binary sample sequence S ($S = \text{'011101010110101111011001'}$) be the target binary sequence under analysis. For instance, we assume that $l = 3$, and the new sequences of S are $S_{k=1} = \langle 011, 101, 010, 110, 101, 111, 011, 001 \rangle$, $S_{k=2} = \langle 011101, 010110, 101111, 011001 \rangle$ and so on. In the example, the total number of items in $S_{k=1}$ is $m_1 = 2+2+1+1+1+1+1 = 8$

and in $S_{k=2}$ is $m_2 = 1 + 1 + 1 + 1 = 4$. Then the serial binary entropy of $S_{k=1}$ is $H_1 = -2 \times (2/8) \log(2/8) - 4 \times (1/8) \log(1/8) = 2.5$ and the serial binary entropy of $S_{k=2}$ is $H_2 = -4 \times (1/4) \log(1/4) = 2$. They would often be turned into standardized entropy, and the final results of standardized entropy are $2.5/\log 8 = 0.8333$ and $2/\log 4 = 1$.

A formula to calculate the serial binary entropy process above was derived as the following. We assume that the binary sequence length is L and the length of fixed-length binary subsequence is l . Then, we define S_k to denote the set in the case of all possible k values, and H_k to denote the serial binary entropy of the binary sequence over the set S_k . And we can calculate H_k in Formula (1).

$$\begin{aligned}
 H_k &= - \sum_{i=1}^{|S_k|} (m_{ik}/\lfloor L/kl \rfloor) \log(m_{ik}/\lfloor L/kl \rfloor) \\
 &= (1/\lfloor L/kl \rfloor) \left[\sum_i \log m_{ik}(\lfloor L/kl \rfloor) - \sum_i m_{ik} \log m_{ik} \right] \\
 &= \log(\lfloor L/kl \rfloor) - (1/\lfloor L/kl \rfloor) \sum_i m_{ik} \log m_{ik}
 \end{aligned} \tag{1}$$

where m_{ik} is the number of occurrences of the i th element in S_k , and it satisfies $\sum_{i=1}^{|S_k|} m_{ik} = \lfloor L/kl \rfloor$.

In this paper, we can use Formula (1) to calculate an entropy vector for a given binary sample sequence and the elements in the list of entropy vector are $\{H_1, H_2, \dots, H_k\}$, for the classification, each H_k is treated as a feature of the binary sample sequence.

4. Classification Architecture Based on Machine Learning. Distinguishing between encrypted and non-encrypted compressed traffic is a binary classification problem. The Support Vector Machine (SVM) and Random Forest (RF) work well in the binary classification problem. Therefore, in this part, we firstly propose an entropy-based feature extraction algorithm for binary sequences based on the serial binary entropy which is defined in Section 3.2. Then based on the feature extraction algorithm proposed, we propose encrypted and non-encrypted compressed traffic classification architecture combined with SVM and RF. We will describe the entropy-based feature extraction algorithm proposed and the identification process architecture for the encrypted and non-encrypted compressed traffic in detail.

4.1. Feature extraction algorithm. In this part, we will describe the complete feature extraction algorithm based on the serial binary entropy for a given binary sequence. We also give a specific example to make the proposed feature extraction algorithm easier to follow.

Algorithm 1 describes the complete feature extraction algorithm for encrypted and non-encrypted compressed traffic sequences. The complete algorithm process can be completed in three phases. The first stage is to initialize the relevant variables and get a fixed-byte-length binary sample sequence. In the second stage, new binary sequences are obtained by sliding the window with a fixed-size and an initialized step. In the third stage, the serial binary entropy is calculated according to a sliding window with fixed-size and step for the sequences obtained in the second stage. Finally, a 8×4 feature matrix F of the given binary sequence can be obtained through Algorithm 1. The specific operation of stages 2 and 3 will be introduced in Algorithm 2 and Algorithm 3.

Algorithm 2 describes how to use a sliding window to transform a given binary sequence into a new binary sequence. In Algorithm 2, the sliding window size is set to 8, which is a byte size. The step size of sliding window is the parameter M . Then we move the

Algorithm 1. Feature extraction algorithm

```

1: Input a fixed-length binary sample sequence  $Q$  (e.g.,  $Q = \langle 10100101 \dots 1101101 \rangle$ )
2: Set the sliding window size sets  $D$  ( $D = \{4, 8, 16, 24\}$ ) and initialize step  $M$  ( $M = 0$ )
3: for  $M = 1$  to 8 do
4:   NewSeq = GetNewSeq( $Q, M$ )
5:   for  $d$  in  $D$  do
6:     if  $d = 4$  then
7:        $k = 0$ 
8:     else
9:        $k = d/8$ 
10:    end if
11:     $H_k = \text{GetEntropy}(\text{NewSeq}, d)$ 
12:    Add  $H_k$  to Feature Matrix  $F$ 
13:  end for
14: end for

```

Algorithm 2. GetNewSeq(Q, M)

```

1: Input a binary sequence  $Q$  and step length  $M$ 
2: Initialize sliding window size  $D$  ( $D = 8$ )
3:  $L \leftarrow$  Get the length of  $Q$ 
4: for  $i = 1$  to  $\lfloor (L - D + M)/M \rfloor$  do
5:   subSeq( $i$ ) = Sliding Window( $Q$ )
6:   Sliding window moves step  $M$ 
7:   Add subSeq( $i$ ) to NewSeq list
8: end for
9: return NewSeq

```

Algorithm 3. GetEntropy(NewSeq, d)

```

1: Input a binary sequence NewSeq and the sliding window size  $d$ 
2: Initialize sliding window step  $M = d$ 
3:  $L \leftarrow$  Get the length of NewSeq
4: for  $i = 1$  to  $\lfloor L/M \rfloor$  do
5:   subSeq( $i$ ) = Sliding Window(NewSeq)
6: end for
7: Calculate  $H$  in Formula (1) for subSeq
8: return  $H$ 

```

sliding window according to the step size M , and merge all the 8-bit binary sequences obtained by the sliding window into a new binary sequence. Algorithm 3 describes how to obtain the entropy feature of the new sequence obtained by Algorithm 2 according to the method of calculating the serial binary entropy explained in Section 3.2.

Figure 1 shows a specific example of extracting feature calculation process based on entropy-based feature extraction algorithm. Assuming that a fixed-length binary sample sequence which is the input Q in Algorithm 1 is $\langle 00001100000101111000000001011010011100000001100 \rangle$. Herein, we give two cases with sliding steps of 7 and 8 which are the value of parameter M in Algorithm 1. As shown in Figure 1, in the case of window size being 8 bits (parameter D in Algorithm 2) and step length being 7 (parameter M in Algorithm 1), the fixed-length binary sample sequence Q is converted to a new sequence $\langle 00001100000010111110000000001011101001111000000 \rangle$ by using Algorithm 2 in Algorithm 1. When the sliding step is 8 (parameter M in Algorithm 1), the fixed-length binary

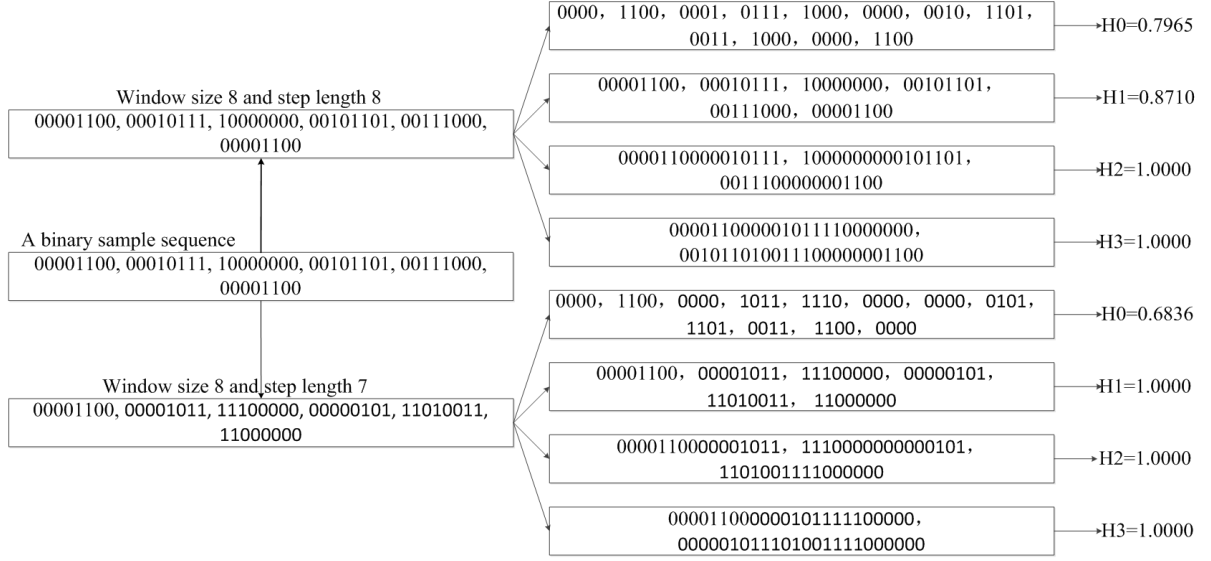


FIGURE 1. A specific example for entropy-based feature extraction algorithm

sample sequence Q does not change by using Algorithm 2 in Algorithm 1. According to the parameter d calculated by the parameter D in Algorithm 1, the sequence obtained in each case can be split into four different-length subsequence sets by using Algorithm 3. At the end of Algorithm 3, the serial binary entropy of the four subsequence sets in each case is calculated. Herein, we only illustrate the serial binary entropy calculation of the subsequence set $\langle 0000, 1100, 0001, 0111, 1000, 0000, 0010, 1101, 0011, 1000, 0000, 1100 \rangle$. The total number of items in this subsequence set is $m = 3+2+1+1+2+1+1+1 = 12$, and the serial binary entropy is $H = -(3/12) \log(3/12) - 2 \times (2/12) \log(2/12) - 5 \times (1/12) \log(1/12) = 2.8554$. The final result of standardized entropy is $H_0 = 2.8554 / \log(12) = 0.7965$. The serial binary entropy calculation process for the remaining subsequence set is similar to this subsequence set. Finally, all H_k are combined to form feature vectors.

4.2. Classification architecture based on machine learning. Support Vector Machine (SVM) was first proposed by Cortes and Vapnik in 1995. It exhibits many unique advantages in solving small sample, nonlinear and high dimensional pattern recognition. Random forest refers to a classifier that uses multiple trees to train and classify samples. It is unexcelled in accuracy among current algorithms and runs efficiently on large data bases. In this paper, we will use SVM and RF to classify the data into two categories: encrypted data and non-encrypted compressed data. The feature space of the data is a 8×4 matrix used as input of the SVM and RF. The output is -1 and 1 , which respectively represent encrypted data and non-encrypted compressed data. The entire identification process architecture is shown in Figure 2.

As shown in Figure 2, the training dataset can be categorized into two cases: the file dataset and the real packet payload dataset. The differences and links between these two datasets will be discussed in detail by the experiment in Section 5. At the pre-processing phase, Packet Capture (PCAP) files or common files are processed and user selects the sequence length according to actual needs. The longer the sequence is, the higher the classification accuracy is, but it also means higher time consumption. Then the processed sequence is used as input of the entropy-based feature extraction algorithm to obtain the feature vector. Finally, the feature vector is input into the corresponding trained machine learning model SVM or RF to obtain the classification category.

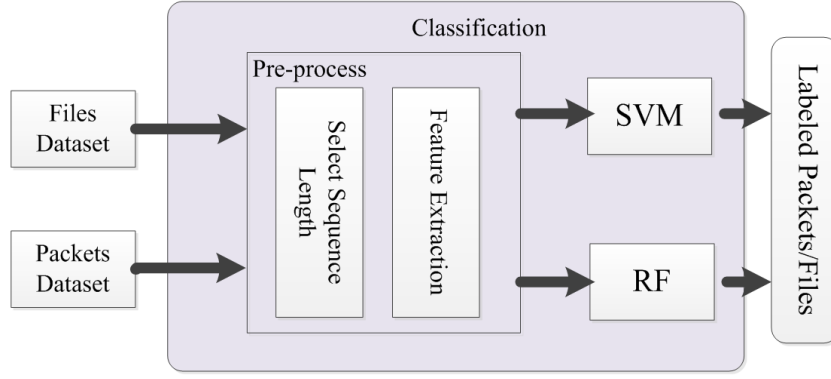


FIGURE 2. Classification architecture for encrypted and non-encrypted compressed data based on SVM and RF

5. Experiment and Result. To demonstrate the advantage of the proposed feature extraction algorithm based on the serial binary entropy for SVM and RF, we compare it with the feature extraction algorithm in [7,9] which is representative in encrypted traffic identification. For a given sequence, the feature vector of the sequence is obtained by computing the consecutive byte entropy of the sequence [7]. For the sake of description, we will abbreviate the consecutive byte entropy based feature extraction algorithm as CBE and abbreviate the proposed feature extraction algorithm based on the serial binary entropy as SBE.

5.1. Description of dataset. The datasets used in the experiment include the file dataset and the real packet payload dataset. The detailed description of each dataset is as follows.

- **Dataset1:** We collected a pool of four classes of the file dataset that are more commonly used: 102MBytes image files (including JPG, GIF, PNG), 100MBytes text files (including TEXT, TXT, PDF), 105MBytes audio files (including MP3, APE, WAV), and 101Mbytes video files (including AVI, MP4, MPG).
- **Dataset2:** The real packet payload dataset is captured from Internet traffic. The collection of non-encrypted compressed traffic is captured through the FTP protocol data transmission between two servers. In order that the non-encrypted compressed traffic is more representative, the data transmitted by the FTP protocol is a mixture of data using multiple compression methods (such as .tar, .zip, and .7z).
- **Dataset3:** The collection of encrypted traffic is from CTU-Normal and ISCX. The encrypted traffic includes HTTPS, VPN and TOR.

5.2. Experimental setup and evaluation metrics. The experimental platform is DELL R720 server which is equipped with CentOS release 73 operate system. The CPU is a 16-cores XeonE5620 2.40GHz, and the memory is 16GB. In all experiments, it is found that the parameter variation of RF has little effect on the classification result. Finally, for convenience of processing, all parameters are considered comprehensively and RF parameters are $n_estimators = 130$ and $min_samples_split = 120$. In this paper, four evaluation metrics were used: accuracy (A), precision (P), recall (R), f1 value (F1). Accuracy was used to evaluate the overall performance of a classifier. Precision, recall and f1 value were used to evaluate performance of every class of traffic.

$$A = \frac{TP + TN}{TP + FP + FN + TN}, \quad P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F1 = \frac{2PR}{P + R}$$

where TP is the number of instances correctly classified as X , TN is the number of instances correctly classified as Not- X , FP is the number of instances incorrectly classified as X , and FN is the number of instances incorrectly classified as Not- X .

5.3. Same content attribute. Dataset1 was used in this section. We obtain encrypted files named Dataset1-ENC and non-encrypted compressed files named Dataset1-GZ, which are generated by files transformation from Dataset1 using the AES encryption algorithms and GZ compression method. The original Dataset1-ENC and Dataset1-GZ are processed with 1024 bytes as a sample. Then the sample number of Dataset_enc and the sample number of Dataset_gz are obtained as shown in Table 2. Finally, the sample number of training set and the sample number of test set are obtained by uniform sampling from Dataset_enc and Dataset_gz. Sampling is to reduce the number of samples and increase the training efficiency. The preprocessed results of the Dataset1 are shown in Table 2. We carry out a grid search on parameter space, and achieve the best classification accuracy with Radial Basis Function (RBF) kernel by $\gamma = 2$, and $C = 32768$ when we use SVM [7].

Figure 3 shows the accuracy of two feature extraction algorithms combined with SVM and RF when the content attributes of encrypted and non-encrypted compressed files are Audio, Image, Text or Video. As we can see from Figure 3, the accuracy of the proposed feature extraction algorithm SBE is better than the feature extraction algorithm CBE in [7,9]. When using the proposed feature extraction algorithm SBE, the classification accuracy of encrypted image files and non-encrypted compressed image files can reach about 72%. However, the classification accuracy of encrypted audio files and non-encrypted compressed audio files is the worst and only reaches more than 65%. Thus, it can be concluded that the classification accuracy of encrypted files and non-encrypted compressed files is very relevant to their content attributes.

TABLE 2. The number of samples that are obtained after Dataset1 processing

Type	Audio	Image	Text	Video
Dataset_enc	108469	104862	103077	103930
Dataset_gz	104698	102302	48873	96283
Training set	18947	18415	17850	18749
Test set	2368	2301	2231	2342

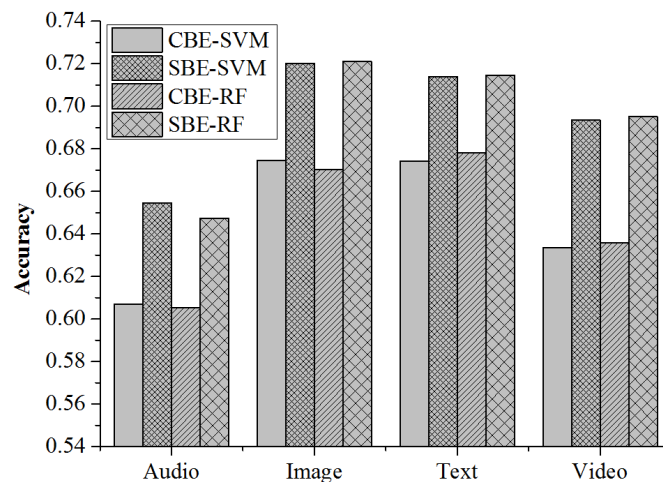
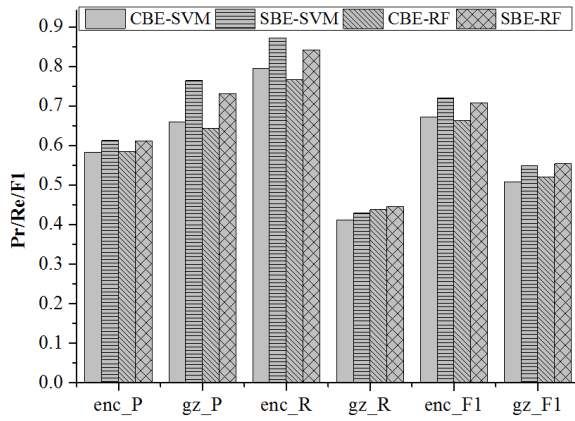
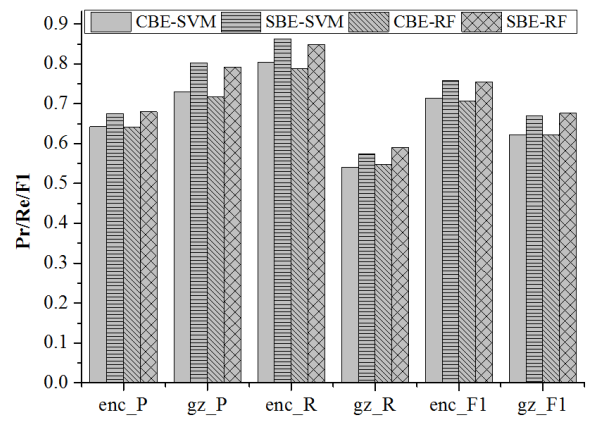


FIGURE 3. The accuracy of two feature extraction algorithms combined with SVM and RF on different contents

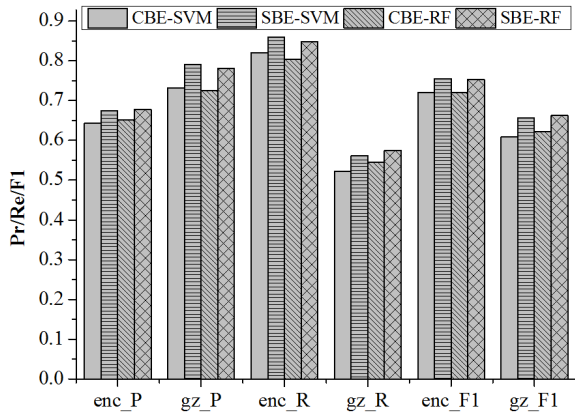
Figure 4 shows the precision, recall and F1 value of two feature extraction algorithms combined with SVM and RF when the content attributes of encrypted and non-encrypted compressed files are Audio, Image, Text or Video. As shown in Figure 4, the proposed feature extraction algorithm SBE is also better than the feature extraction algorithm CBE in [7,9] in precision, recall and F1 value. From Figure 4, the classification precision of the encrypted files is low, and the recall is high. However, the classification result of the non-encrypted compressed files is just the opposite. This phenomenon indicates that some non-encrypted compressed files are incorrectly identified as encrypted files. Especially in the classification experiment of encrypted audio files and non-encrypted compressed audio files, the recall of non-encrypted compressed audio files is only about 40%.



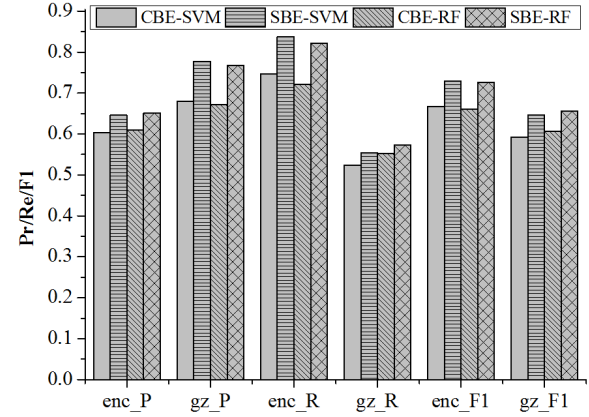
(a) The non-encrypted compressed and encrypted audio files classification



(b) The non-encrypted compressed and encrypted image files classification



(c) The non-encrypted compressed and encrypted text files classification



(d) The non-encrypted compressed and encrypted video files classification

FIGURE 4. The precision, recall and F1 value of two feature extraction algorithms combined with SVM and RF on different contents

5.4. Different content attributes. In Section 5.3, we studied the classification performance of the proposed algorithm SBE under the condition of encrypted and non-encrypted compressed content attribute being the same. However, this situation is unlikely to occur in real life. In this section, we study the classification performance of the proposed feature extraction algorithm SBE in the case where the encrypted and non-encrypted compressed content attributes are different. The dataset used in this experiment is still Dataset_enc and Dataset_gz in Section 5.3. By exchanging encrypted and non-encrypted compressed

files samples of audio, image, text, and video, 12 sets of experimental dataset are finally obtained. The sample number of training set and the sample number of test set are also obtained by uniform sampling from Dataset_enc and Dataset_gz. The two classes of samples in each dataset each account for approximately 50%. The preprocessed results of the Dataset_enc and Dataset_gz are shown in Table 3. We also carry out a grid search on parameter space, and achieve the best classification accuracy with Radial Basis Function (RBF) kernel by $\gamma = 2$, and $C = 32768$ when we use SVM.

TABLE 3. The number of samples that are obtained after Dataset_enc and Dataset_gz processing

	Training set	Test set
Audio_enc vs Image_gz	18735	2341
Audio_enc vs Text_gz	18329	2291
Audio_enc vs Video_gz	19151	2393
Audio_gz vs Image_enc	18627	2328
Audio_gz vs Text_enc	18468	2308
Audio_gz vs Video_enc	18545	2317
Image_enc vs Text_gz	18009	2251
Image_enc vs Video_gz	18831	2353
Image_gz vs Text_enc	18256	2281
Image_gz vs Video_enc	18333	2290
Text_enc vs Video_gz	18672	2333
Text_gz vs Video_enc	17927	2240

Figure 5 shows the performances of two feature extraction algorithms combined with SVM and RF when the content attributes of encrypted and non-encrypted compressed files are different. According to the classification results of all the situations, whether the content attributes of the encrypted and non-encrypted compressed files are the same or not has a slight impact on the classification accuracy. As shown in Figure 5, the proposed feature extraction algorithm SBE is still better than the feature extraction algorithm CBE in [7,9]. When the content attributes of encrypted and non-encrypted compressed files are different, the identification accuracy is also about 60% to 75%. From Figures 5(d), 5(e), and 5(f), when the content attribute of one class is the non-encrypted compressed audio file, the classification accuracy is relatively low, which is only 60% or so. And the lowest recall, which is only more than 40%, also indicates that the non-encrypted compressed audio files are difficult to identify. It is consistent with the difficulty of identifying the non-encrypted compressed audio files in Section 5.3. Contrast to the low classification accuracy of non-encrypted compressed audio files, the classification accuracy of other content attribute files is basically over 70%. Of course, the difference in content attribute does not change the phenomenon that some non-encrypted compressed files are mistakenly identified as encrypted files.

5.5. Real traffic. The experiments in Sections 5.3 and 5.4 both validate the feasibility of the proposed classification algorithm SBE on the datasets obtained by encrypting and compressing files. However, the real network traffic is not the same as the ideally constructed dataset. In this section, we will use the real network traffic Dataset2 and Dataset3 to verify the effectiveness of the proposed classification algorithm SBE. Herein, the raw packets are processed to obtain the TCP or UDP flow. A flow is defined as all packets that have the same 5-tuple, i.e., source IP, source port, destination IP, destination port and transport protocol. By sampling method, only first n bytes ($n = 900, 1024, 1156$,

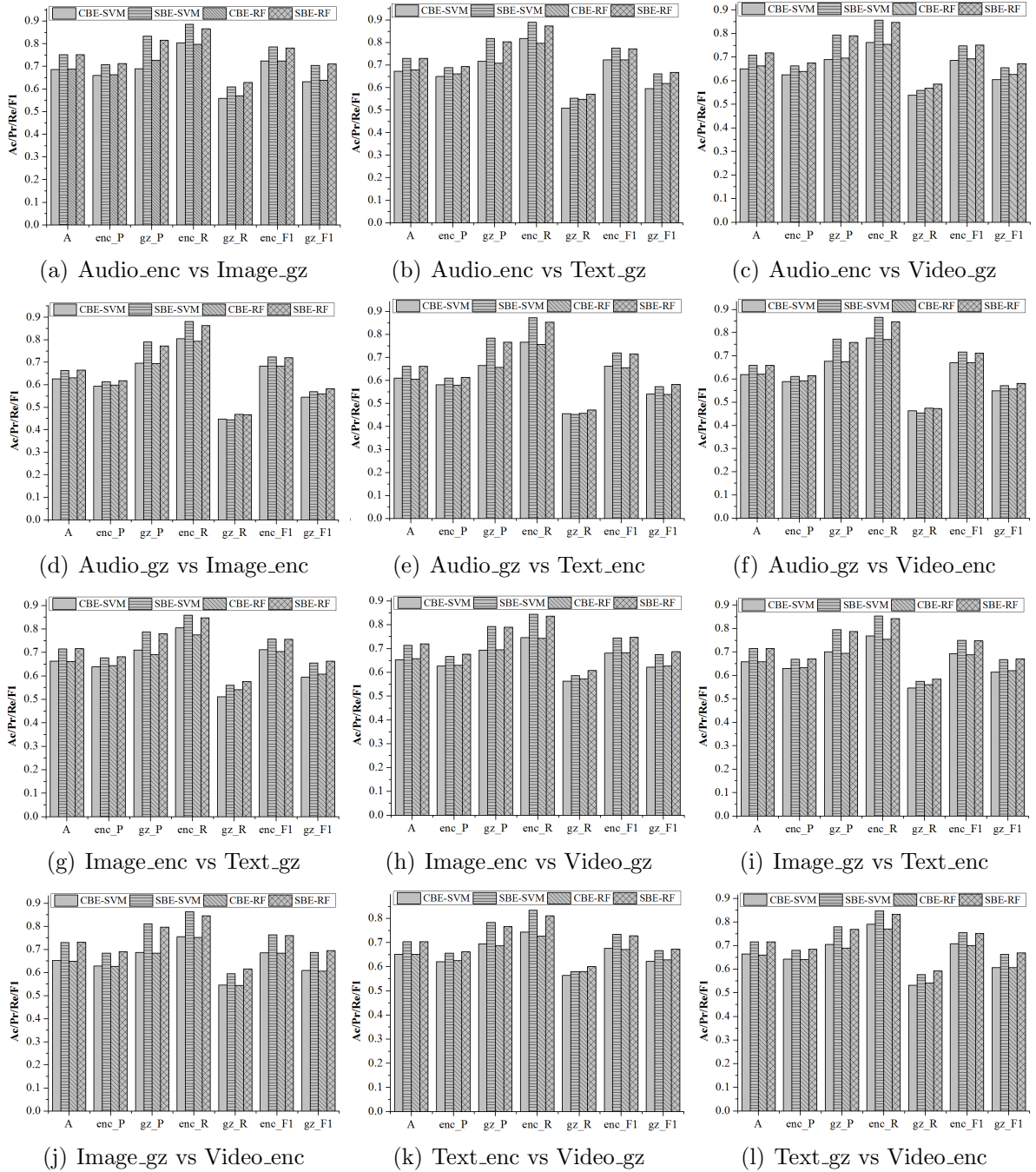
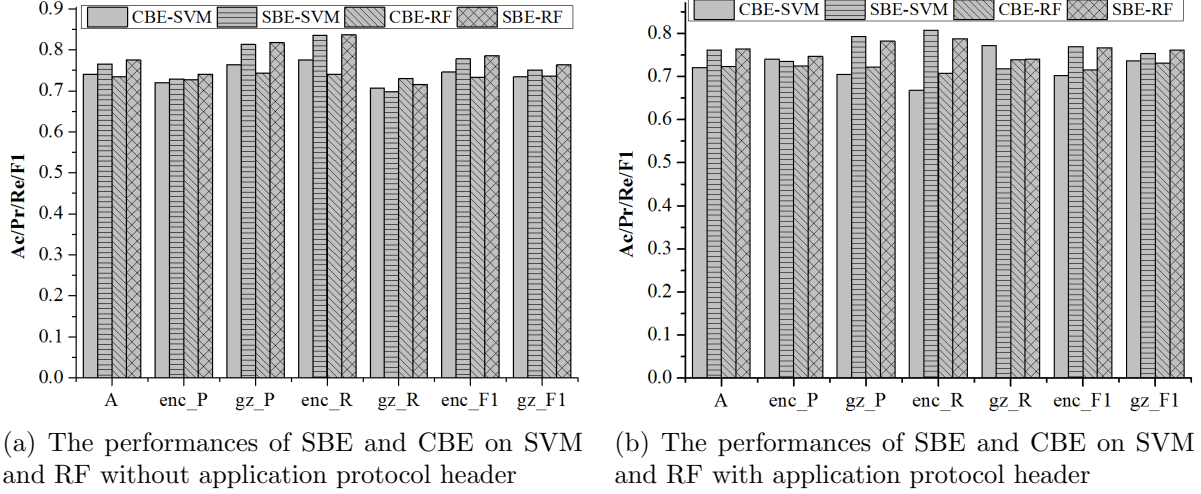


FIGURE 5. The performances of two feature extraction algorithms on SVM and RF for different content attributes of encrypted and non-encrypted compressed files

1296, 1444 in this paper) of each packet payload in flow are used. If packet payload size is larger than n bytes, it is trimmed to n bytes. If packet payload size is shorter than n bytes, the 0 is added in the end to complement it to n bytes. The sample number of training set and the sample number of test set are obtained by sampling as shown in Table 4. The training and test sets include two classes: encrypted traffic samples and non-encrypted compressed traffic samples. We carry out a grid search on parameter space, and achieve the best classification accuracy with Radial Basis Function (RBF) kernel by $\gamma = 50$, and $C = 1000$ when we use SVM.

TABLE 4. The sample number of training set and test set for encrypted traffic and non-encrypted compressed traffic

Length (Bytes)	900	1024	1156	1296	1444
Training set	26698	26698	26698	26698	26698
Test set	3337	3337	3337	3337	3337



(a) The performances of SBE and CBE on SVM and RF without application protocol header

(b) The performances of SBE and CBE on SVM and RF with application protocol header

FIGURE 6. The performances of two feature extraction algorithms for encrypted traffic and non-encrypted compressed traffic with application protocol header or not

Figure 6 shows the classification results of real encrypted traffic and non-encrypted compressed traffic. Figure 6(a) shows the classification performance of using 1024-byte packet payload without application protocol header information and Figure 6(b) shows the classification result of using 1024-byte packet payload that contains the application protocol header information. From Figures 6(a) and 6(b), it can be seen that the classification performance of non-using application protocol header information is well matched with that using application protocol header information. Even though the application protocol header information is not encrypted, it does not help to improve the classification accuracy. In follow-up experiments, for the convenience of data processing, the packet payload will include the application protocol header information. As seen in Figure 6(a), the recall of non-encrypted compressed traffic with using SBE is lower than that uses CBE, but the precision is opposite. This explains that the SBE algorithm combined with SVM or RF can more accurately identify non-encrypted compressed traffic and reduce the probability that encrypted traffic is misidentified as non-encrypted compressed traffic. As shown in Figure 6(b), when using the SBE-SVM algorithm, the precision of encrypted traffic and the recall of non-encrypted compressed traffic are not as good as the CBE-SVM algorithm. It demonstrates that when using the SBE-SVM algorithm, there will be some non-encrypted compressed traffic that is misidentified as encrypted traffic. Apart from this, the proposed feature extraction algorithm SBE has achieved better results than CBE whether application protocol header information is used or not.

As shown in Figure 7, with the packet payload length used in the experiment gradually increasing, the classification accuracy is continuously improving whether the feature extraction algorithm is SBE or CBE. As seen in Figure 7, when the packet payload lengths are 900, 1024, 1156 and 1296 bytes, the performance of the proposed feature extraction

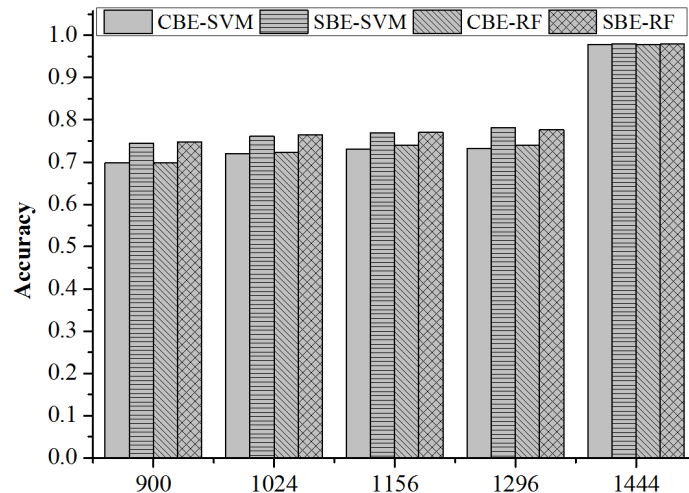


FIGURE 7. The performances of SBE and CBE for encrypted traffic and non-encrypted compressed traffic classification with different payload lengths

algorithm SBE is better than the CBE algorithm in [7,9]. In the best case, the proposal outperforms 4.9% the CBE algorithm in terms of classification accuracy. In the worst case, they are well-matched in classification accuracy when the packet payload length is 1444 bytes. The longer the packet payload length is, the more information that can be extracted. So that both SBE and CBE can perform better in classification accuracy. When the packet payload length is 1444 bytes, the classification accuracy of SBE-RF and SBE-SVM can reach 97.90%.

5.6. Analysis and discussion. The feature extraction algorithm CBE was firstly proposed in [7]. Authors use CBE-SVM algorithm to classify and identify content attributes of network traffic. Authors discuss and analyze the performance issues of the CBE in distinguishing encrypted from compressed traffic. In [9], authors used the feature extraction algorithm CBE with CART and SVM, but did not optimize the algorithm. Encrypted and non-encrypted compressed traffic classification problem is not studied. The latest paper on the optimization of feature extraction algorithms CBE is [8]. A Monte Carlo simulation method used to estimate the error of π value was proposed to supplement the feature space in [7]. However, there is no experiment for distinguishing between encrypted and non-encrypted compressed traffic in [8]. Given that the improved method in [8] is not based on entropy, in order to make a more scientific and fair comparison, we chose the feature extraction algorithm CBE in [7,9]. The CBE algorithm is more similar to the proposed feature extraction algorithm. From the above experiments, we can find that the proposed feature extraction algorithm SBE performs better than the feature extraction algorithm CBE in [7,9] when the packet payload length is short. Moreover, the proposed feature extraction algorithm has better performance regardless of whether it is combined with SVM or RF. When the payload length increases, the more information is available and the advantage of the proposed feature extraction algorithm is no longer outstanding. However, it still is well-matched in classification accuracy compared with CBE.

6. Conclusions and Future Work. The identification of encrypted traffic as a challenging topic in traffic management and security detection has attracted the attention of researchers. Since non-encrypted compressed data and encrypted data both have random feature, distinguishing them becomes more difficult. A novel entropy-based feature

extraction algorithm was proposed in this paper. For a fixed-length binary sequence, it obtains new sequences through different sampling and combinations, so as to change the randomness of the original sequence. The idea behind this is that encrypted data is highly random, and therefore random feature should not change regardless of the different combinations. For the lossless compressed data, there is obvious deviation from randomness in the randomness test [25]. Therefore, the recombination of compressed data will change its randomness. Finally, the experiments for the SBE on the real traffic are verified. The performance of the proposed entropy-based feature extraction algorithm SBE is better than the CBE in [7,9].

In the future, on the basis of the proposed feature extraction algorithm, further improvements and optimizations will be made. It will hopefully reach a goal that is small computing space and high classification accuracy. Then it can perform well in future real-time network packet analysis system environment [26].

Acknowledgment. This work was supported by “The Next-Generation Broadband Wireless Mobile Communications Network” National Science and Technology of Major Projects (No. 2017ZX03001019). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

- [1] A. Baer, P. Casas, A. D’Alconzo, P. Fiadino, L. Golab et al., DBStream: A holistic approach to large-scale network traffic monitoring and analysis, *Computer Networks*, vol.107, pp.5-19, 2016.
- [2] J. Brown, M. Anwar and G. Dozier, An evolutionary general regression neural network classifier for intrusion detection, *IEEE International Conference on Computer Communication and Networks*, pp.1-5, 2016.
- [3] M. Iwamoto, S. Oshima and T. Nakashima, A malware detection method based on OC-SVM focusing on features of PDF files, *ICIC Express Letters*, vol.11, no.11, pp.1611-1618, 2017.
- [4] X. Yun, Y. Wang, Y. Zhang and Y. Zhou, A semantics-aware approach to the automated network protocol identification, *IEEE/ACM Trans. Networking*, vol.24, no.1, pp.583-595, 2016.
- [5] C. Tian and L. Xuan, An optimized solution of application layer protocol identification based on regular expressions, *The 18th Asia-Pacific Network Operations and Management Symposium*, pp.1-4, 2016.
- [6] S. Yan and B. Subir, Characterization of traffic analysis based video stream source identification, *IEEE International Conference on Advanced Networks and Telecommunications Systems*, pp.1-6, 2015.
- [7] Y. Wang, Z. Zhang, L. Guo and S. Li, Using entropy to classify traffic more deeply, *Proc. of IEEE International Conference on Networking, Architecture, and Storage*, pp.45-52, 2011.
- [8] G. Cheng and Y. Chen, Identification method of encrypted traffic based on support vector machine, *Journal of Southeast University (Natural Science Edition in Chinese)*, vol.47, no.4, pp.655-659, 2017.
- [9] A. R. Khakpour and X. A. Liu, An information-theoretical approach to high-speed flow nature identification, *IEEE/ACM Trans. Networking*, vol.21, no.4, pp.1076-1089, 2013.
- [10] P. Velan, M. Cermák, P. Čeleda and M. Drašar, A survey of methods for encrypted traffic classification and analysis, *International Journal of Network Management*, vol.25, no.5, pp.355-374, 2015.
- [11] A. Dainotti, A. Pescapé and K. C. Claffy, Issues and future directions in traffic classification, *IEEE Network*, vol.26, no.1, pp.35-40, 2012.
- [12] M. Finsterbusch, C. Richter, E. Rocha, J. A. Müller and K. Hänßgen, A survey of payload-based traffic classification approaches, *IEEE Communications Surveys and Tutorials*, vol.16, no.2, pp.1135-1156, 2014.
- [13] T. Karagiannis, K. Papagiannaki and M. Faloutsos, BLINC: Multilevel traffic classification in the dark, *Computer Communication Review*, vol.35, no.4, pp.229-240, 2005.
- [14] P. Perera, Y. C. Tian, C. Fidge and W. Kelly, A comparison of supervised machine learning algorithms for classification of communications network traffic, *Lecture Notes in Computer Science*, vol.10634, pp.445-454, 2017.
- [15] M. P. Singh, G. Srivastava and P. Kumar, Internet traffic classification using machine learning, *International Journal of Database Theory and Application*, vol.9, no.12, pp.45-54, 2016.

- [16] Z. X. Chen, Z. S. Liu, L. Z. Peng, L. Wang and L. Zhang, A novel semi-supervised learning method for Internet application identification, *Soft Computing*, vol.21, no.8, pp.1963-1975, 2017.
- [17] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas and J. Lloret, Network traffic classifier with convolutional and recurrent neural networks for Internet of Things, *IEEE Access*, vol.5, pp.18042-18050, 2017.
- [18] W. B. Pan, G. Cheng, X. J. Guo and S. X. Huang, Review and perspective on encrypted traffic identification research, *Journal on Communications*, vol.37, no.9, pp.154-167, 2016.
- [19] J. Sherry, C. Lan, R. A. Popa and S. Ratnasamy, BlindBox: Deep packet inspection over encrypted traffic, *Proc. of the ACM Conf. on Special Interest Group on Data Communication*, pp.213-226, 2015.
- [20] M. Korczyński and A. Duda, Markov chain fingerprinting to classify encrypted traffic, *Proc. of IEEE Conf. on Computer Communications*, pp.781-789, 2014.
- [21] B. Zaho, H. Guo, Q. R. Liu and J. X. Wu, Protocol independent identification of encrypted traffic based on weighted cumulative sum test, *Journal of Software*, vol.24, no.6, pp.1334-1345, 2013.
- [22] W. Wang, M. Zhu, J. L. Wang, X. W. Zeng and Z. Z. Yang, End-to-end encrypted traffic classification with one-dimensional convolution neural networks, *IEEE International Conference on Intelligence and Security Informatics: Security and Big Data*, pp.43-48, 2017.
- [23] G. Xiong, W. T. Huang, Y. Zhao, M. Song, Z. Z. Li and L. Guo, Real-time detection of encrypted thunder traffic based on trustworthy behavior association, *Communications in Computer and Information Science*, vol.320, pp.132-139, 2013.
- [24] G. L. Sun, Y. B. Xue, Y. F. Dong, D. S. Wang and C. L. Li, A novel hybrid method for effectively classifying encrypted traffic, *IEEE Global Telecommunications Conference*, pp.1-5, 2010.
- [25] W. Chang, B. Fang, X. Yun, S. Wang and X. Yu, Randomness testing of compressed data, *Journal of Computing*, vol.2, 2010.
- [26] S. J. Dang, X. Liu, X. K. Wang and C. M. Liu, Design of real-time data collection and analysis system based on spark streaming, *Network New Media*, vol.6, no.5, 2017 (in Chinese).