IJNS
International Journal of Network Security

ISSN 1816-353X (Print)
ISSN 1816-3548 (Online)

Vol. 24, No. 4 (July 2022)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan, R.O.C.
Email: mshwang@asia.edu.tw

International Journal of Network Security is published both in traditional paper form (ISSN 1816-353X) and in Internet (ISSN 1816-3548) at http://ijns.jalaxy.com.tw

PUBLISHER: Candy C. H. Lin

Volume: 24, No: 4 (July 1, 2022)

International Journal of Network Security

Adversarial Examples Generation Method for Chinese Text Classification

En-Hui Xu[1], Xiao-Lin Zhang[1], Yong-Ping Wang[1], Shuai Zhang[1], Li-Xin Liu[2], and Li Xu[3]
(Corresponding author: Xiao-Lin Zhang)

School of Information Engineering, Inner Mongolia University of Science and Technology[1]
Baotou 014010, China
School of Information, Renmin University of China[2]
Beijing 100872, China
Baotou Medical College, Inner Mongolia University of Science and Technology[3]
Baotou 014010, China
(Email: zhangxl@imust.edu.cn)

(Received Nov. 22, 2021; Revised and Accepted Apr. 2, 2022; First Online Apr. 10, 2022)

Abstract

Aiming at the problem that DNNs-based text classification systems are vulnerable to adversarial example attacks, a method of adversarial example generation for Chinese text classification, WordHit, is proposed. In this method, we use the morphological and phonological features of Chinese characters to establish a pool of similar characters and homophones, find important words or phrases that affect classification by removing non-contributing clauses and calculating word importance scores and design a modification strategy that combines word sound and word form to generate adversarial examples to achieve a black-box attack on Chinese text classification models. The word-CNN model and the BiLSTM model are used to verify the effectiveness and versatility of different classification tasks. It is proved that the adversarial example generated by this method can be effectively transferred to the BERT model and the actual deployed sentiment analysis system.

Keywords: Adversarial Examples; Black-box Attack; Chinese Character Characteristics; Chinese Text Classification; Deep Neural Networks

1 Introduction

In recent years, Deep Neural Networks (DNNs) [19] have been widely used in many fields such as computer vision, speech recognition and natural language processing, however, Szegedy *et al.* [15] found that these neural network models are exceptionally vulnerable to adversarial example attacks. To address the security problem, many adversarial example generation methods such as FGSM (Fast Gradient Sign Method) [5], Deepfool [10], C&W [1], and PGD (Project Gradient Descent) [9] have been proposed one after another. However, most of these methods are targeted at images [17], and the discrete properties of text and metrics different from those of images make these methods not directly applicable to text. In the textual domain, Papernot *et al.* [11] first proposed to generate adversarial examples, where the authors used FGSM to find adversarial perturbations to modify the word vector, however, the modified word vector may not have words corresponding to it, so the authors used a specific dictionary to select words to replace the original words. Although the mapping problem is solved, more words unrelated to the original word are introduced, resulting in grammatical errors. In addition, Samanta *et al.* [14] used FGSM to locate important words and created a candidate pool for each word in advance, and then modified the first k important words using three strategies: insertion, replacement and deletion. However, there may be some important words without candidates in the actual input. The above two methods are performed in a white-box scenario [20], and Gao *et al.* [3] studied the adversarial example generation method in a black-box scenario [4, 6, 7] and proposed the DeepWordBug algorithm. The algorithm uses the output of the model to find the keywords in the original text by the word importance calculation function, and then generates adversarial examples using insertion, deletion, replacement and exchange of characters. Ren *et al.* [13] proposed a greedy algorithm for word-level attacks by first determining the keywords to be replaced by Probability Weighted Word Saliency (PWWS) to determine the order of keywords to be replaced, and then use WordNet to find synonyms to generate adversarial examples. Similar to Ren *et al.* Zang *et al.* [18] proposed a word-level attack algorithm based on sememes, which can generate more diverse adversarial examples by finding the words with the same sememes corresponding to each word in the original sample through HowNet and then using a particle swarm

algorithm to optimize the combination of candidate words in the discrete space.

The above adversarial example generation methods are designed based on English text and have not been studied for Chinese text, and certain character-level modification strategies are not applicable to Chinese. Wang *et al.* [16] first proposed an adversarial example generation method for Chinese, and they used a pre-trained substitution model and word importance calculation function to determine the keywords to be replaced, and used homophones for replacement. Since the method only uses the phonetic features of Chinese characters for keyword replacement, the features are not fully utilized and the modification strategy is relatively single. Based on the above research work, we propose an adversarial example generation method WordHit for Chinese text, which constructs a candidate pool by analyzing the word form and phonetic features of Chinese characters, then calculates the important words affecting the model classification by a new screening algorithm of important words or phrases, and finally modifies the important words using a modification strategy that combines word sound and word form to generate adversarial examples. A word-level black box attack against Chinese text under a multi-scene classification task is effectively implemented. The main contributions of this paper are follows.

1) WordHit, an adversarial example generation method for Chinese text, is proposed to generate adversarial examples by only slightly modifying the original text without the need to understand the target model parameters. It can interfere with classification tasks of multiple scenarios, such as sentiment analysis, spam classification and news classification.

2) Candidate pools of similar characters and homophones are established for Chinese texts, and the candidate words maintain high semantic similarity with the original words, which can effectively ensure the quality and diversity of the generated adversarial examples.

3) A new filtering method for important words or phrases is designed, which can effectively identify the key words affecting the model decisions under different classification tasks, reduce the modification rate, and generate adversarial examples at a smaller cost.

4) Experiments on real datasets using WordHit to attack word-CNN and BiLSTM in the sentiment analysis task reduce the model accuracy to below 50%, and the attack effectiveness is better than baseline methods. In the spam classification task and news classification task, it also reduces the model accuracy to around 50%, demonstrating the generality of the adversarial example generation algorithm. In addition, the transferability of adversarial examples is successfully exploited to attack BERT and two actually deployed sentiment analysis systems.

The rest of this article is organized as follows. The text adversarial example is described and defined in detail in Section 2. The adversarial example generation method WordHit is introduced in Section 3. In Section 4, experiments are conducted on four real datasets. Finally, the conclusion of the paper is given in Section 5.

2 Text Adversarial Examples

Given a data set $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ with n texts and a set of corresponding n labels $\mathcal{Y} = \{y_1, y_2, \ldots, y_n\}$. A pre-trained natural language classification model F, which needs to learn the mapping $f : \mathcal{X} \to \mathcal{Y}$ from the input text $x \in \mathcal{X}$ to the label $y \in \mathcal{Y}$. Finally, it can classify the original input text x to the true label y_{true} as much as possible, as shown in Equation (1):

$$\arg\max_{y_i \in \mathcal{Y}} P\left(y_i \mid x\right) = y_{\text{true}} \,. \tag{1}$$

Under normal circumstances, an adversarial example x' is generated by adding a small disturbance r to x. The adversarial example will cause model F to give a wrong label, as in Equation (2):

$$\arg\max_{y_i \in \mathcal{Y}} P\left(y_i \mid x'\right) \neq y_{\text{true}} \,. \tag{2}$$

At the same time, the disturbance is required to be imperceptible to the human eye ,which means it will not cause significant changes in semantics, ensuring that humans can still understand the meaning of the original text. So the adversarial example can be defined as in Equation (3):

$$\begin{gathered} x' = x + r, \quad \|r\|_p < \epsilon, \\ \arg\max_{y_i \in \mathcal{Y}} P\left(y_i \mid x'\right) \neq \arg\max_{y_i \in \mathcal{Y}} P\left(y_i \mid x\right). \end{gathered} \tag{3}$$

In Equation (3), $\|r\|_p$ defined in Equation (4) uses p-norm to represent the constraint on perturbation r, and L_0, L_2 and L_∞ are commonly used.

$$\|r\|_p = \left(\sum_{i=1}^{n} |w_i^* - w_i|^p\right)^{\frac{1}{p}}. \tag{4}$$

In Equation (4), the original input text is expressed as $x = w_1 w_2 \ldots w_i \ldots w_n$, where $w_i \in \mathbb{D}$ is a word and $\mathbb{D}$ is a dictionary of words. In order to make the perturbation small enough to be undetectable by humans, the textual adversarial examples need to satisfy word constraints, grammar constraints, and semantic constraints. The word constraint requires that the modified word cannot be the wrong word, the grammar constraint is to ensure the grammatical correctness of the adversarial example, and the semantic constraint ensures that the generated adversarial example should retain the original semantic information. To satisfy the above constraints, homophones and similar characters are used to achieve modification of Chinese text, and a maximum modification threshold σ is

set to constrain the modification magnitude of the adversarial examples. Thus, the effective adversarial example x' can be further expressed Equation (5).

$$F(x') \neq F(x), \text{Cost}(x', x) \leq \sigma. \quad (5)$$

where Cost $(\cdot)$ is the cumulative frequency of text modification.

3 WordHit

DNNs-based text classification system classifies text based on its features, not every word plays the same role in the classification label, some key words closely affect the classification result, and changing key words can largely change the original classification label. In this paper, we visit the target model to locate the keywords that affect the classification. The specific process is shown in Figure 1.

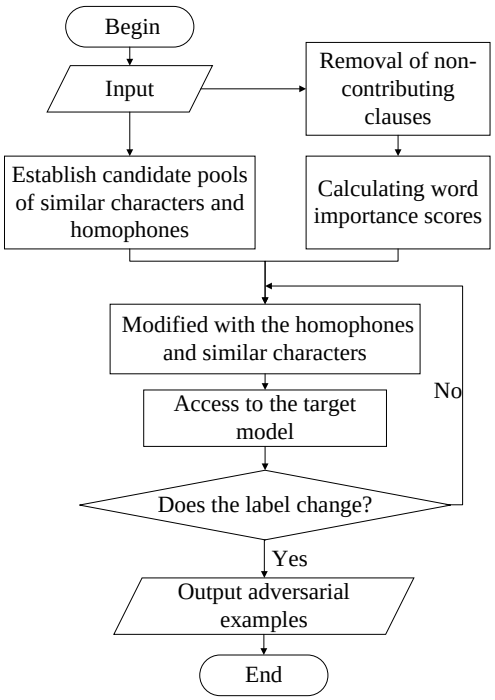


Figure 1: Adversarial example generation process

The process is described as follows.

① Establish candidate pools of similar characters and homophones: analyze the character shape and phonetic characteristics of Chinese text, and establish a candidate pool of similar characters and homophone candidates for each Chinese character.

② Removal of non-contributing clauses: The original text is divided into clauses, and the result of removing each clause in turn is input to the target model to obtain a score relative to the correct label. By calculating the difference with the original score, the clauses that do not contribute to the current classification label are removed.

③ Calculating word importance scores: The retained clauses are divided into words and filtered out stop words, each word is marked as UNK in turn, and then the marked results are input to the target model, and the important words or phrases that affect the classification of the model are ranked by calculating word importance scores.

④ Modified with the homophones and similar characters: one important word or phrase is selected at a time in the order of ranking, and it is modified into the corresponding word in the candidate pool using the word sound and word form modification strategy.

The modified results are input to the target model to obtain the classification labels, and if the labels do not change, the execution continues ④ until the predicted labels of the target model change or reach the modification threshold, and the final generated adversarial examples are output.

3.1 Establish Candidate Pools of Similar Characters and Homophones

The word level-based text adversarial example generation method usually requires a candidate pool, and the quality of the candidate pool determines the quality of the adversarial example to a certain extent. The degree of similarity between Chinese characters can be measured by the sameness or similarity of "sounds", or the similarity of "shapes". Taking the glyph and phonetic features of Chinese characters as the analysis object, a candidate pool is generated for each Chinese character.

3.1.1 Candidate Pool of Similar Characters

In order to analyze the similarity relationship between characters more flexibly, a simple character similarity comparison method is designed. Commonly used Chinese characters are obtained from GB1312 area code, and under a given font, character bitmaps are obtained and rendered into fixed-size pictures with grayscale values between 0 and 255 for each pixel point. Each picture is converted into a vector according to the size of the grayscale value, and the degree of similarity between Chinese characters is analyzed by comparing the Euclidean distance between two vectors. Following this strategy, for each Chinese character, the top n characters closest to it can be found. Table 1 shows the morphological similarities corresponding to some Chinese characters when n is taken as 1,2,3.

In order to maximize the perceptual similarity between the generated candidate pool of similar character and the original character, the optimal candidate for each character is obtained by taking $n = 1$ and adding it to the candidate pool of similar character. Considering that the

Table 1: Candidate Pool of Similar Characters and Homophonic Characters Corresponding to Some Chinese Characters. Ori.(Original words). BS(The best similar characters). TC(Traditional Chinese). SW(Splitting words).CPoSC(Candidate pool of similar characters). CPoH(Candidate pool of homophones)

Ori.	Similar words			CPoSC			CPoH		
	$n=1$	$n=2$	$n=3$	BS	TC	SW	Homophones of similar shape	Pinyin	Other homophones
评	怦	坪	抨	怦	評	讠平	坪枰呯玶蚜胖鲆鲆平苹萍	Ping	凭屏瓶
假	徦	瑕	蝦	瑕	/	亻叚	徦婽椵叚瘕	Jia	甲钾胛贾玾价加家驾
睬	睐	踩	眯	睐	/	目采	踩婇採綵綵埋啋采菜彩	Cai	蔡材财猜
惠	恵	蕙	崽	恵	/	/	恵蕙澮憓德橞璓蟪	Hui	会回汇荟绘慧
鲜	鲆	鲑	蚌	鲆	鮮	鱼羊	鲜薢灕薢廯	Xian	先仙闲显险掀现限线
真	直	具	其	直	/	/	禛嫃眞逭	Zhen	针阵珍贞甄臻祯桢侦镇

simplified and traditional forms of a Chinese character can represent the same semantic information, the corresponding traditional form of the character is added to the candidate pool. The splitting of a left-right structure does not affect human reading, so if the character has a left-right structure, the split character is added to the candidate pool as a candidate. The results of the morphological candidate pool are shown in Table 1.

3.1.2 Candidate Pool of Homophones

Unlike similar characters, a Chinese character often has many homophones. The homophones are further divided into two parts: homophones of similar shape and other homophones. The homophones of similar shape can have a certain degree of morphological similarity while ensuring similar pronunciation, while the other homophones only require similar pronunciation. If the candidate pool does not reach the set capacity, the pinyin and other homophones of the word are added to the candidate pool. In the experiment, the maximum capacity of the homophone candidate pool was set to 15, and n was set to 20. The results of the homophone candidate pool are shown in Table 1.

3.2 Important Words or Phrases Filtering Algorithm

To ensure the readability and validity of the text obtained after modification, modifying important words or phrases in the text and controlling the magnitude of the changes are the basic strategies of text adversarial generation. Which are the important words and how to locate them are the problems to be solved by this algorithm.

3.2.1 Remove Non-contributing Clauses

An input sample often contains multiple sentences, but not every sentence contributes to the classification label. In order to locate key words more efficiently, meaningless sentences need to be eliminated. First, the original sample x is divided into n clauses using punctuation marks to obtain $x = \{s_1, s_2, \ldots, s_n\}$. For the i-th clause in the sequence, the confidence difference (Delete Score, DS) between the input after removing the clause and the original input is calculated in turn as shown in Equation (6).

$$DS(s_i) = F(s_1, \ldots, s_{i-1}, s_i, s_{i+1}, \ldots, s_n) - F(s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n) \tag{6}$$

If $DS(s_i) \leq 0$, then it means that s_i does not contribute to the current classification label of the sequence and it is removed from the original input. After n visits to the target model, the final filtered sequence X is obtained.

3.2.2 Calculating Word Importance Scores

The sequence X obtained by the first filtering operation contains all the clauses that contribute to the current classification label. Then, the important words or phrases that affect the classification in all clauses are found by calculating the word importance scores. All the sentences in X are divided and the meaningless stop words are filtered out to obtain: $X = \{w_1, w_2, \ldots, w_i, \ldots, w_n\}$. After marking each word in turn as an unknown character UNK, it is input to the target model, and the difference between the original score returned by the target model and the current score is counted as the importance score of the word, as shown in Equation (7).

$$S(w_i) = F(w, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots, w_n) - F(w_1, \ldots, w_{i-1}, \text{UNK}, w_{i+1}, \ldots, w_n) \tag{7}$$

Finally, each word is sorted by importance score from highest to lowest to get the list of words to be modified X'.

3.3 Modified with the Homophones and Similar Characters

Combining the established candidate pool with the list of words to be modified X', a modification strategy $\boldsymbol{T}$ that combines word sound and word form is proposed. The specific description is as follows.

(1) Select a word or phrase to be modified in the vocabulary list X' in order.

(2) The corresponding candidate pool of similar characters and candidate pool of homophones are selected with probabilities of k $(k \in [\ 0,\ \ 1\ ])$ and $1-k$, respectively.

(3) If the pool selected is the candidate pool of similar characters, the candidate words in the pool are randomly selected for replacement; If the pool selected is the candidate pool of homophones, the homophones of similar shape is first selected with probability j $(j \in [\ 0,\ \ 1\ ])$, and the pinyin and other homophones are selected with probability $1-j$, and then the words in the selected range are randomly selected for replacement.

The complete WordHit algorithm is shown in Algorithm (1).

Algorithm 1 WordHit

Input: Text x; Target classification model F; Modify strategy $\boldsymbol{T}$; Modify threshold δ
Output: Adversarial example x'

Begin
$x = \{s_1, s_2, \ldots, s_n\}$
for $i = 1, \cdots, n$ **do**
  $DS(s_i) = F(s_1, \ldots, s_{i-1}, s_i, s_{i+1}, \ldots, s_n) - F(s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$
  if $DS(s_i) \leq 0$ **then**
    $x \leftarrow$ Delete s_i from x
  end if
end for
$X \leftarrow x$
$X = \{w_1, w_2, \ldots, w_i, \ldots, w_n\}$
for i = 1, $\cdots$, n **do**
  $S(w_i) = F(w, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots, w_n) - F(w_1, \ldots, w_{i-1}, \mathrm{UNK}, w_{i+1}, \ldots, w_n)$
end for
$X' \leftarrow$ sort w_i by descending $S(w_i)$
for w_i in X' **do**
  $w_i^* = \boldsymbol{T}(w_i)$
  $x' \leftarrow$replace w_i with w_i^* in x'
  if $F(x') \neq F(x)$ and Cost $(x', x) \leq \sigma$ **then**
    return x'
  end if
end for

4 Empirical Evaluation

4.1 Experiment Setting

Four publicly available datasets were used for validation and performance evaluation of the WordHit algorithm, and the specific data information is shown in Table 2. word-based CNN (word-CNN) [12] and Bi-directional LSTM (BiLSTM) [8] were used as the target models for the experiments. Among them, the word-CNN consists of a 300-dimensional embedding layer, three convolutional layers and a fully connected layer, and the convolutional layer consists of 256 convolutional kernels of size 2,3,4 with a step size of 1. The BiLSTM consists of a 300-dimensional embedding layer, a bidirectional LSTM layer and a fully connected layer, and the forward and backward directions of the bidirectional LSTM layer consist of 64 LSTM units, respectively. The forward and backward directions of the bi-directional LSTM layer are composed of 64 LSTM cells respectively.

4.2 Comparison of Experimental Methods

Validation of the effectiveness of the WordHit algorithm on the sentiment analysis task and comparison of two baseline algorithms: DeepWordBug [6] and WordHanding [16]. The performance of the WordHit algorithm was evaluated separately on the spam classification task and the news classification task. In the experiments, the maximum modification threshold δ was set to 11 for the news classification task and 30 for the other tasks, and WordHit selected the strategy of phonological modification for important words or phrases, setting k to 0.5 and j to 0.6. The experimental results are shown in Table 3 and Table 4.

For both datasets of the sentiment analysis task, the model accuracy reduction exceeds that of the baseline method, indicating that WordHit outperforms DeepWordBug and WordHanding in misleading classifiers with better attacks. As shown by the experimental results of the spam classification task and the news classification task, the WordHit algorithm succeeds in both classification tasks, bringing down the classification accuracy to about 50%, proving its generality under multi-scene classification tasks.

In order to verify the relationship between the accuracy of model detection and the modification threshold, the same experimental setup as the WordHanding algorithm is maintained, and 1000 samples with length greater than 120 words are selected from the two sentiment analysis datasets respectively, and the adversarial examples are generated by adjusting different modification thresholds to investigate the effect of different thresholds on the effectiveness of the generated adversarial examples. Figure 2 and Figure 3 show the variation curves of the detection accuracy of the two models with the modification threshold on the Ctrip dataset and the JingDong(JD) dataset, respectively. The accuracy of the model detection gradually decreases as the modification threshold increases, i.e., the text has enough modification space to modify the keywords affecting the classification. Compared with the baseline method, the WordHit algorithm can reduce the model detection accuracy significantly by modifying only a few words, and the attack effect has leveled off when the number of modified words reaches 15, indicating that WordHit can minimize the modification of the original sample and ensure the text readability.

Table 2: Statistics on the datasets

Dataset	Classes	Train samples	Test samples	Average words	Task
Ctrip	2	12000	3000	132	Sentiment analysis
JD	2	35000	5000	36	Sentiment analysis
Spam	2	90000	10000	51	Spam classification
THUCNews	10	90000	10000	19	News classification

Table 3: Validation of the algorithm WordHit on sentiment analysis tasks

(a) word-CNN

Dataset	Ori_acc(%)	Baselines				Ours	
		DeepWordBug		WordHanding		WordHit	
		Accuracy(%)	Reduction	Accuracy(%)	Reduction	Accuracy(%)	Reduction
Ctrip	92.03	77.48	14.55	69.53	22.50	41.07	50.96
JD	91.46	75.04	16.42	70.43	21.03	48.42	43.04

(b) BiLSTM

Dataset	Ori_acc(%)	Baselines				Ours	
		DeepWordBug		WordHanding		WordHit	
		Accuracy(%)	Reduction	Accuracy(%)	Reduction	Accuracy(%)	Reduction
Ctrip	92.03	77.48	14.55	69.53	22.50	41.07	50.96
JD	91.46	75.04	16.42	70.43	21.03	48.42	43.04

4.3 Adversarial Examples Quality Measurement

To measure the quality of the generated adversarial examples, the WMD (Word Mover's Distance) is used to measure the similarity between adversarial examples and the original samples. The smaller the WMD, the higher the semantic similarity. 1000 data and their corresponding adversarial examples are randomly selected from two datasets for the experiment. The proportion of WMDs in different intervals is shown in Figure 4.

WordHit algorithm has the largest percentage on the interval of 0-0.2, which indicates that the adversarial examples generated by WordHit algorithm have higher semantic similarity with the original samples. The WMD of the adversarial examples generated by WordHit algorithm in the interval of 0-0.6 accounts for 78.3% of the data, which is higher than the baseline method, indicating that most of the generated adversarial examples have higher quality and can better retain semantic information. Table 5 shows examples of the adversarial examples generated using the WordHit algorithm, which can be seen that the generated adversarial examples retain the original semantics and can be understood by humans.

4.4 Human Evaluation

To further explore the impact of the adversarial examples generated by the WordHit algorithm on human reading, we conducted a human evaluation. The main two aspects of the evaluation were to assess the accuracy of human classification of the generated adversarial examples and to assess the naturalness of the adversarial examples from the perspective of human perception. This was done by randomly selecting 100 clean and corresponding confrontation samples from the two sentiment analysis datasets, disrupting the samples and giving them to volunteers for classification, and giving them a likelihood score between 1 and 5, the higher the score, the more likely the human was to write the article. A total of six volunteers participated in the experiment, and Table 6 shows the evaluation results.

As can be seen from Table 6, although adversarial examples make the model misclassify, the human classification effect is still very good, and the difference between the classification accuracy and that of the clean samples is less than 5%, indicating that the adversarial examples generated by the WordHit algorithm retain the semantic information of the original text better and do not affect people's reading comprehension of the text content. In terms of the naturalness score, although it is lower than that of the clean sample, the difference is small, indicating that the naturalness of the adversarial example is within the range acceptable to humans.

4.5 Transferability Assessment

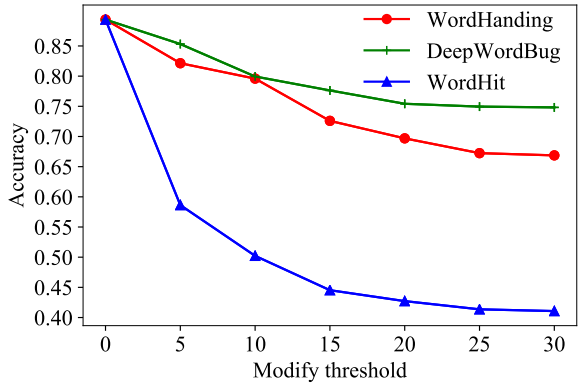
In the field of text classification, the transferability of adversarial examples means that the adversarial examples generated against one classification model can be used to attack other models as well. Using the transferability of adversarial examples, adversarial examples can be gener-

Table 4: Evaluating WordHit performance on spam classification and news classification tasks

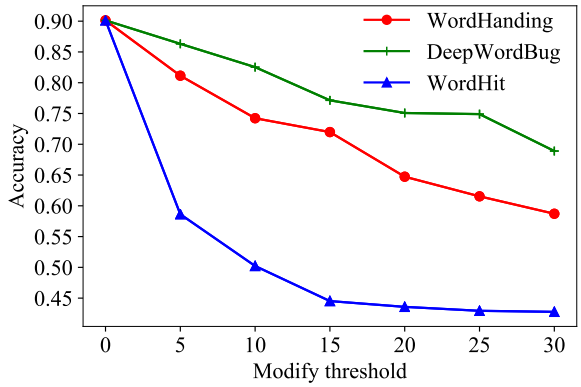
Dataset	Model	Ori_acc(%)	WordHit	
			Accuracy(%)	Reduction
Spam	word-CNN	99.86	49.95	49.91
	BiLSTM	99.76	49.78	49.98
THUCNews	word-CNN	90.89	53.40	37.49
	BiLSTM	90.64	52.79	37.85

Table 5: Examples of original samples and generated adversarial examples

Original sample	Label	Adversarial example	Label
房间巨小, 电视成了摆设, 开不了, 服务员态度冷漠, 不睬我, 以后不会再进这家酒店.	Negative	房间剧 xiao, 电视成了摆设, 开丕了, 服务员态度泠莫. bu 睐我, 以后布绘再进这家酒店.	Positive
东西非常实惠, 快递真给力, 昨天下的单今天就到了, 新包装颜色鲜艳, 好评.	Positive	东西非常湿蕙, 快递直铪厉, 昨天下的单今天就到了, 新包装颜色鲜燕,女子評.	Negative
用完特别容易痒, 跟以前屈臣氏的根本不一样, 绝对的假货! 贪小便宜吃大亏!	Negative	用完特别容易养, 跟以前屈臣氏的裉奔步一样, 绝对的嘏货!谈小便宜吃大兮!	Positive
酒店脚摩很有特色, 住客还给打折, 房间装修尚可, 位置稍偏.	Positive	酒店脚摩恨油牪 寺涩, 住客还給 Da 浙, 房间装修伤渴, 位置稍偏.	Negative

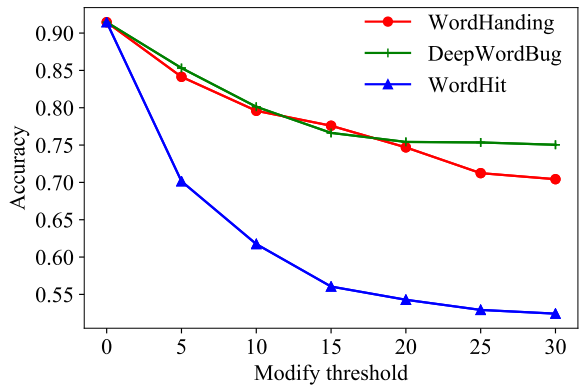


(a) word-CNN

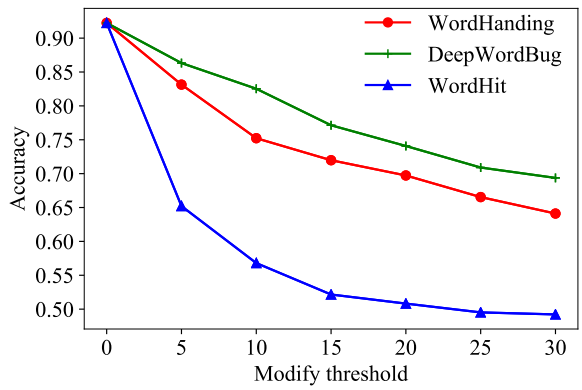


(b) BiLSTM

Figure 2: The variation curve of detection accuracy with modification threshold for the adversarial example of Ctrip review dataset



(a) word-CNN



(b) BiLSTM

Figure 3: The variation curve of detection accuracy with modification threshold for the adversarial example of JD review dataset

Table 6: Comparison with human evaluation.

Dataset	Model	Examples	Accuracy of model(%)	Accuracy of human(%)	Score[1-5]
Ctrip	word-CNN	Original	97.00	97.00	4.70
		Adversarial	18.00	93.00	3.83
	BiLSTM	Original	93.00	97.00	4.70
		Adversarial	21.00	93.00	3.83
JD	word-CNN	Original	96.00	98.00	4.50
		Adversarial	23.00	95.00	3.95
	BiLSTM	Original	95.00	98.00	4.50
		Adversarial	25.00	95.00	3.95

Table 7: Results of adversarial examples generated using BiLSTM model to attack other models/systems

Dataset	Model/Cloud Platform	Ori_acc(%)	WordHit	
			Accuracy(%)	Reduction
Ctrip	word-CNN	92.03	57.37	34.66
	BERT	91.60	58.40	33.20
	Tencent Cloud	87.67	61.40	26.27
	Baidu AI	88.43	63.10	25.33
JD	word-CNN	91.46	61.37	30.09
	BERT	92.52	62.78	29.74
	Tencent Cloud	89.42	65.44	23.98
	Baidu AI	88.90	63.78	25.12

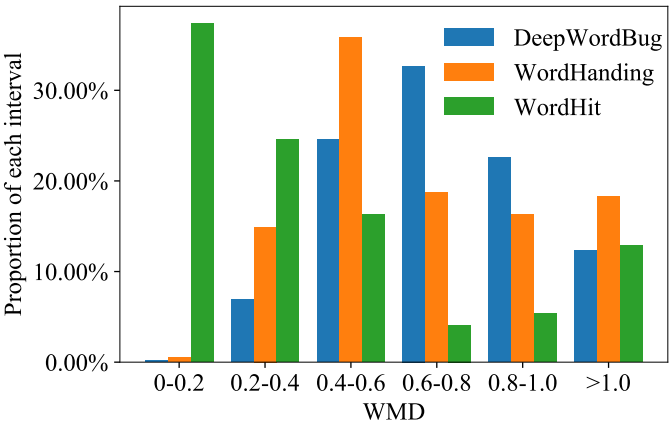


Figure 4: Proportion of the number of samples in different WMD distance intervals to the total samples

ated on alternative models to achieve a black-box attack on the target model. To evaluate the transferability of the adversarial examples generated by the WordHit method, in addition to the word-CNN model and the BiLSTM model, the BERT [2] model, and two actual deployed sentiment analysis systems (Tencent Cloud Sentiment Analysis System and Baidu Cloud Sentiment Analysis System) were additionally introduced. Table 7 and Table 8 show the results of adversarial example attacks on other models (systems) generated using the BiLSTM model and the word-CNN model, respectively. The results show that the adversarial examples generated by the WordHit algorithm for both models can be effectively transferred to the other four models (systems), causing the classification accuracy of the word-CNN, BILSTM and BERT models to drop by about 30% and the accuracy of the two sentiment analysis systems to drop by about 25%.

4.6 Adversarial Training

Adversarial training is a technique to improve model robustness by adding adversarial examples to the training set and repeating the training to improve model robustness, which is commonly used in image classification and can also be used as a means to enhance model generalization in natural language processing tasks. To analyze the effect of adversarial training on classification accuracy, 5000 data items are randomly selected from the Ctrip hotel review dataset, and adversarial examples are generated as the ensemble $\mathbb{A}$ using WordHit on the BiLSTM model. Several adversarial examples are randomly selected from the ensemble $\mathbb{A}$ and added to the original training set to evaluate the classification accuracy of the original test set and the classification accuracy of the adversarial example test set.

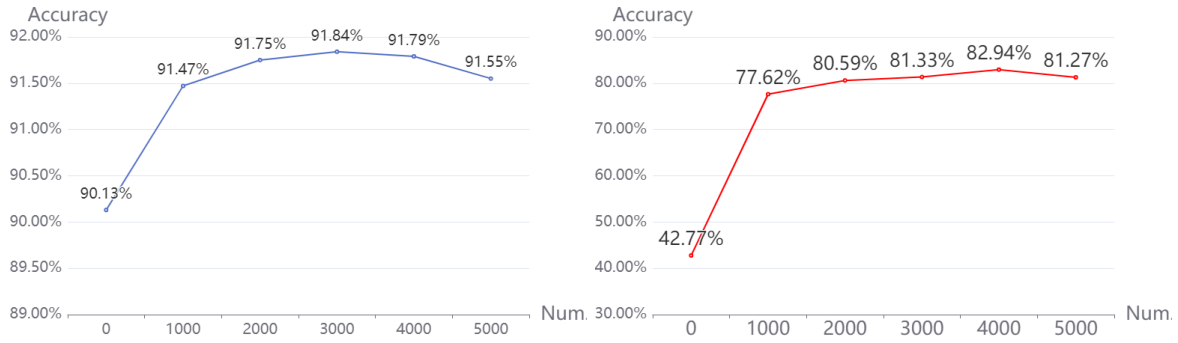
The data in Figure 5(a) show that the adversarial training helps to improve the accuracy of the classification model. Figure 5(b) illustrates that when more and more adversarial examples are involved in the training, the robustness of the model steadily improves and the classification accuracy can be improved to over 80%.

5 Conclusions

Adversarial example generation for Chinese text classification models is important for evaluating and improving

Table 8: Results of adversarial examples generated using word-CNN model to attack other models/systems

Dataset	Model/Cloud Platform	Ori_acc(%)	WordHit	
			Accuracy(%)	Reduction
Ctrip	BiLSTM	90.13	54.33	35.80
	BERT	91.60	60.57	31.03
	Tencent Cloud	87.67	59.83	27.84
	Baidu AI	88.43	62.50	25.93
JD	BiLSTM	92.24	61.52	30.72
	BERT	92.52	63.48	29.04
	Tencent Cloud	89.42	62.38	27.04
	Baidu AI	88.90	64.22	24.68



(a) Accuracy of original sample test set

(b) Accuracy of adversarial example test set

Figure 5: Effect of adversarial training on the classification accuracy of original and adversarial examples

Chinese text classification systems. In the WordHit algorithm, we use the word form and speech features of Chinese characters to construct a candidate pool, find the keywords or phrases affecting classification by important sentence screening and word importance calculation, and design a modification strategy for adversarial example generation, finally realizing a word-level black box attack against Chinese text classification models under a multi-scene classification task. The experimental results show that the adversarial examples generated by the WordHit algorithm effectively reduce the classification accuracy while retaining the semantic information of the original text with good readability. The vulnerability of current Chinese text classification models is revealed by attacking the BERT model and the actual deployed sentiment analysis system using the transferability of adversarial examples, and the impact of adversarial training on the classification accuracy of the original test set and the adversarial example test set is further analyzed. Therefore, we use this as the basis for our work on the defense of such attack algorithms in the next phase, and explore more robust deep learning models and methods.

Acknowledgments

This work was partially supported by the Natural Science Foundation of China (No.61562065) and the Natural Science Foundation Project of Inner Mongolia (No.2019MS06001 and No.2019MS06036). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017.

[2] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[3] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, "Black-box generation of adversarial text sequences to evade deep learning classifiers," in *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2018, pp. 50–56.

[4] S. Garg and G. Ramakrishnan, "Bae: Bert-based adversarial examples for text classification," *arXiv preprint arXiv:2004.01970*, 2020.

[5] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *Computer Science*, 2014.

[6] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits, "Is bert really robust? a strong baseline for natural language

attack on text classification and entailment," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 05, 2020, pp. 8018–8025.

[7] L. Li, R. Ma, Q. Guo, X. Xue, and X. Qiu, "Bert-attack: Adversarial attack against bert using bert," *arXiv preprint arXiv:2004.09984*, 2020.

[8] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, 2011, pp. 142–150.

[9] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017.

[10] S. M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Computer Vision & Pattern Recognition*, 2016.

[11] N. Papernot, P. McDaniel, A. Swami, and R. Harang, "Crafting adversarial input sequences for recurrent neural networks," in *MILCOM 2016-2016 IEEE Military Communications Conference*. IEEE, 2016, pp. 49–54.

[12] A. Rakhlin, "Convolutional neural networks for sentence classification," *GitHub*, 2016.

[13] S. Ren, Y. Deng, K. He, and W. Che, "Generating natural language adversarial examples through probability weighted word saliency," in *Proceedings of the 57th annual meeting of the association for computational linguistics*, 2019, pp. 1085–1097.

[14] S. Samanta and S. Mehta, "Towards crafting text adversarial samples," *arXiv preprint arXiv:1707.02812*, 2017.

[15] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *Computer Science*, 2013.

[16] W. Wang, R. Wang, L. Wang, and B. Tang, "Adversarial examples generation approach for tendency classification on chinese texts," *Ruan Jian Xue Bao/J. Softw.*, vol. 30, no. 8, pp. 2415–2427, 2019.

[17] Q. Yaguan, L. Hongbo, J. Shouling, Z. Wujie, W. Shuhui, Y. Bensheng, T. Xiangxing, and L. Jingsheng, "Adversarial example generation based on particle swarm optimization," *Journal of Electronics and Information*, vol. 41, no. 7, pp. 1658–1665, 2019.

[18] Y. Zang, F. Qi, C. Yang, Z. Liu, M. Zhang, Q. Liu, and M. Sun, "Word-level textual adversarial attacking as combinatorial optimization," *arXiv preprint arXiv:1910.12196*, 2019.

[19] H. Zhao, Y. K. Chang, and W. J. Wang, "Research on robustness of deep neural networks based data preprocessing techniques," *International Journal of Network Security*, vol. 24, no. 2, pp. 243–252, 2022.

[20] X. Zheng, J. Zeng, Y. Zhou, C. J. Hsieh, M. Cheng, and X. J. Huang, "Evaluating and enhancing the robustness of neural network-based dependency parsing models with adversarial examples," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 6600–6610.

Biography

En-Hui Xu is a graduate student in the School of Information Engineering, Inner Mongolia University of Science & Technology. His research interests include machine learning security and natural language processing.

Xiao-lin Zhang received the Ph.D. degree from the Northeastern University of China, Shenyang, in 2006. She is a professor in the School of Information Engineering, Inner Mongolia University of Science & Technology. Her research interests include database theory and machine learning security, cloud computing, and social network privacy protection.

Yong-ping Wang received a Master's degree from the Wuhan University of Technology in 2010. She is a lecturer in the School of Information Engineering, Inner Mongolia University of Science & Technology. Her research interests are mainly in the field of object detection and data privacy protection.

Shuai Zhang is a graduate student in the School of Information Engineering, Inner Mongolia University of Science & Technology. His research interests include machine learning security and computer vision.

Li-xin Liu is a PhD candidate at the Renmin University of China and the Member of China Computer Federation. Her main research interests include privacy protection and blockchain.

Li Xu received a Master's degree from the Southwest University of Science and Technology in 2009. Her research interest is mainly in the area of image processing and data privacy protection.

Collusion Resistance CP-ABE Scheme with Accountability, Revocation and Privacy Preserving for Cloud-based E-health System

Zhenhua Liu[1,2], Yingying Ding[1,2], Ming Yuan[1,2], and Baocang Wang[3]
(Corresponding author: Yingying Ding)
School of Mathematics and Statistics, Xidian University[1]
Xi'an 710071, P.R. China
(Email: 2318326053@qq.com)
State Key Laboratory of Cryptology, P. O. Box 5159[2]
Beijing 100878, P.R. China
State Key Laboratory of Integrated Services Networks, Xidian University[3]
Xi'an 710071, P.R. China
(Received Oct. 6, 2021; Revised and Accepted Mar. 13, 2022; First Online Apr. 11, 2022)

Abstract

Cloud-based E-health systems can support users to store their health records in the cloud for better care, and ciphertext-policy attribute-based encryption (CP-ABE) with particular functionalities can enhance secure sharing. However, most traceable and revocable schemes only considered user traceability, and the revoked users could access data by conspiring with an unrevoked user. This paper presents a collision resistance CP-ABE scheme with accountability, revocation, and policy hiding. A user's decryption key is associated with the path in a binary tree and a self-selected secret value. By auditing the leaf node value and the secret value, a user or the authority is determined to take responsibility for a compromised key. Then using the binary tree can implement user revocation, which ensures collision avoidance and backward security. Furthermore, the security of the proposed scheme is proven, and the performance analysis indicates that the proposed scheme is efficient.
Keywords: Accountable Authority; Attribute-based Encryption; Collusion Resistance; User Revocation

1 Introduction

With the progress of cloud computing technology, cloud storage system has provided great convenience for users in data storage and sharing [28]. Hence, individuals and enterprises tend to outsource their data to the cloud for reducing storage costs. Due to the above characteristics, cloud storage system is appropriate for electronic health (E-health) system [24]. In order not to impede data sharing and ensure data security, a fine-grained access control system over encrypted data is urgently needed.

Attribute-based encryption is first described by Sahai and Waters [1], which can implement "one to many" access control. However, some malicious users existing in the system may reveal their decryption keys for some benefits. Since a decryption key is related to a user's attribute set, the user who had the same attribute set and divulged the decryption key cannot be identified. In order to settle the above problem, the concept of traceable CP-ABE [5] was proposed. Furthermore, a malicious user should be revoked immediately. In addition, most of the existing revocable CP-ABE schemes [12–14,25,26] supposed that the authority is fully trusted. However, the authority generates the decryption keys for all users and could also abuse the keys. Therefore, there are great expectations for a revocable CP-ABE with the authority accountability for the E-health system.

Moreover, considering that access policies in the form of plaintext are coupled with ciphertext and stored directly in the cloud, it is inevitable to reveal the sensitive information of patients in the E-health system [4]. In order to enhance the privacy protection of users in the E-health system, the CP-ABE schemes that can implement hidden policy were proposed [11, 17].

1.1 Related Works

In order to track down malicious users, Li *et al.* presented the first accountable CP-ABE scheme [6] that supports AND-gate policies. To enhance the expressiveness of access policies, Liu *et al.* constructed a white-box traceable CP-ABE scheme supporting any monotone access structures [5] and a black-box traceable CP-ABE scheme [7] in 2013. However, Liu *et al.*'s two accountable schemes

were structured by utilizing bilinear groups of composite order that were inefficient. Later, a white-box traceable CP-ABE schemes were proposed by Ning *et al.* [8], which based on bilinear groups of prime order.Unfortunately, the above schemes only offered the solution of user key abuse. Nevertheless, the authority can generate decryption keys for illegal users without the threats of being caught. Out of that reason, Ning *et al.* designed the first accountable authority CP-ABE scheme [9] based on the bilinear groups of composite order, which can support flexible access policies. Later, Zhang *et al.* described a CP-ABE scheme [10] based on LSSS that supports the authority accountability, and Li *et al.* presented an accountable authority CP-ABE scheme [11] with hidden policy by utilizing the bilinear groups of prime order. But none of the above schemes took into account user revocation. Considering, the privilege for the baleful user to decrypt a ciphertext should be revoked immediately. Thus, a secure revocation CP-ABE scheme needs to be proposed to revoke the malicious user.

To revoke the illegal user, Hur *et al.* constructed an indirect user revocation CP-ABE scheme [13] based on a more expressive access tree structure, where a secret key or decryption key includes two parts. Unfortunately, Hur *et al.*'s scheme suffered from collusion attacks. In order to avoid user collusion, Li *et al.* [14] proposed a CP-ABE scheme, in which group secret key and private key are bound together by embedding the same random values in two keys. To improve the expressiveness, Lee *et al.* put forward a revocable CP-ABE scheme [15] that can support LSSS access policy. Recently, Ning *et al.* [16] structured two schemes with authority accountability and user revocation: ATER-CP-ABE and ATIR-CP-ABE, where the former realizes revocation by the revocation list and the later implements revocation by key update, and Han *et al.* proposed a scheme [17] with traceability and user revocation. However, their schemes cannot implement the backward security, which means that the previous ciphertext cannot be decrypted by the revoked user [4]. At the same time, the above schemes fail to consider such a problem that the access structure stored directly in the cloud could reveal user sensitive information in the E-health system.

Although numerous of ABE schemes have been presented to protect users' privacy data, sensitive information carried by access policies in ciphertext will still expose users' privacy. To prevent the above problem, many hidden-policy CP-ABE schemes [18,19] were constructed.

1.2 Our Motivation and Contributions

Han *et al.* [17] proposed a multi-purpose CP-ABE scheme, which implemented user accountability, user revocation and hidden policy. However, their scheme could encounter with the collusion attacks that the revoked users can cooperate with the unrevoked users to decrypt the ciphertext and required the authority is full trusted. Though the schemes of Ning *et al.* [16] and Li *et al.* [11] considered the semi-trusted authority, the former failed to guarantee the backward security, while the later cannot adopt the flexible LSSS access policies and achieve user revocability.

1.2.1 Comments on Han *et al.*'s Scheme

In Han *et al.*'s scheme [17], suppose that there are two users Alice and Bob, where the former has the decryption key $(K' = c,\ K = g^{\frac{\alpha}{a+c}} \cdot h^r,\ L = g^r,\ L' = g^{a \cdot r},\ \{K_\tau = g^{s_\tau \cdot r} \cdot u^{-(a+c)\cdot r}\}_{\tau \in I_S},\ K_A = g^{\frac{r}{x_{i_d}}},\ \{x_i\}_{i \in path(i_d)}, \mathcal{S})$ and the latter has the decryption key $((K')^* = c^*,\ K^* = g^{\frac{\alpha}{a+c^*}} \cdot h^{r^*},\ L^* = g^{r^*},\ (L')^* = g^{a \cdot r^*},\ \{K^*_\tau = g^{s_\tau \cdot r^*} \cdot u^{-(a+c^*)\cdot r^*}\}_{\tau \in I^*_S},\ K_B = g^{\frac{r^*}{x_{i^*_d}}},\ \{x_i\}_{i \in path(i^*_d)}, \mathcal{S}^*)$. They want to access the ciphertext $(C = m \cdot e(g,g)^{\alpha s},\ C_0 = g^s,\ C'_0 = g^{a \cdot s},\ \{C_{i,1} = h^{\lambda_i} \cdot u^{k_i},\ C_{i,2} = g^{-k_i \cdot t_{\rho(i)} + \lambda_i},\ C_{i,3} = g^{k_i}\}_{i \in [1,l]},\ \{T_j = y_j^s\}_{j \in cover(\mathcal{R})}, \mathcal{R}, \overline{W})$, where the set $cover(\mathcal{R})$ means the minimum cover set associated with the revocation list $\mathcal{R}$. Alice is an unrevoked user, but her attribute set $\mathcal{S}$ does not meet $\overline{W}$. On the other hand, Bob's attribute set $\mathcal{S}^*$ meets $\overline{W}$, but as a revoked user, Bob cannot also decrypt the ciphertext since there are no elements in $cover(\mathcal{R}) \cap path(i^*_d)$. Furthermore, none of $\{x_i\}_{i \in path(i^*_d)}$ can be used to decrypt the ciphertext. Although they fail to decrypt the ciphertext individually, they can successfully decrypt the ciphertext if they combine their respective decryption keys. Since Alice is not be revoked, she can obtain the node $j \in cover(\mathcal{R}) \cap path(i_d)$ and give x_j to Bob. Once Bob obtains x_j, and since his attribute set $\mathcal{S}^*$ meets $\overline{W}$, he can decrypt the ciphertext as follows:

1) Firstly, use x_j to compute $\frac{x_{i^*_d}}{x_j}$, then compute $B = e(K_B, T_j)^{\frac{x_{i^*_d}}{x_j}} = e(g,g)^{r^* \cdot s}$.
2) Secondly, calculate E, F, and D:

$$\begin{aligned} E =& [e((L^*)^{(K')^*} \cdot (L')^*, C_{i,1}) \cdot e(L^*, C_{i,2}) \\ & \cdot e(K^*_{\rho(i)}, C_{i,3})] \\ =& e(g,h)^{(a+c^*)\cdot r^* \cdot \lambda_i} \cdot e(g,g)^{r^* \cdot \lambda_i}, \\ F =& \prod_{i \in I} (E)^{c_i} = e(g,h)^{(a+c^*)\cdot r^* \cdot s} \cdot e(g,g)^{r^* \cdot s}, \\ D =& e(K^*, C_0^{(K')^*} \cdot C'_0) \\ =& e(g,g)^{\alpha \cdot s} \cdot e(g,h)^{(a+c^*)\cdot r^* \cdot s}. \end{aligned}$$

3) Finally, recover the message $m = \frac{C \cdot F}{D \cdot B}$.

Moreover, Bob can decrypt the ciphertext without anyone's help, just by changing the decryption algorithm a little. B is computed as follows:

$$B = e(K_B, C_0)^{x_{i^*_d}} = e(g,g)^{r^* \cdot s},$$

and E, F, D are calculated by the same way as before. Thus Bob can successfully get the plaintext. Due to the aforementioned flaws, Han *et al.*'s scheme cannot support the backward security and forward security, where the forward security means the revoked user cannot decrypt the ciphertext in the future [4].

1.2.2 Our Contributions

Inspired with Han *et al.*'s scheme [17] and Li *et al.*'s scheme [11], an accountable and revocable CP-ABE (AR-CP-ABE) scheme with privacy protection is proposed, which can provide the authority and user accountability, the direct user revocation, the backward security, and the partial hidden policy. The main contributions and techniques are as follows:

1) **Authority and user accountability.** We construct a CP-ABE scheme with the authority and the user accountability based on the bilinear groups of prime order and flexible LSSS access policies. In the proposed scheme, the full decryption key of a user contains the partial keys generated by the authority based on the attribute set and the path in a binary tree that one leaf node value corresponds to one user, which is used to trace the user, and a secret value chosen by the user, which is used to ultimately determine whether the compromised key was generated by the user or the authority.

2) **User collusion avoidance.** In the proposed scheme, we bind the values of the node on the user path to the value in connection with the user's identity, which can avoid the collusion between the unrevoked user and the revoked user. Furthermore, we define the collusion resistance security model between the revoked user and the unrevoked user, and give a detailed proof that our scheme can resist the user collusion attack.

3) **Backward security and forward security.** In Han *et al.*'s scheme [17], the revoked user can obtain the plaintext with knowing the value x_{i_d} of node, which could break the backward security and forward security. To solve this problem, we embed x_{i_d} into the key instead of sending $\{x_i\}_{i\in path(i_d)}$ directly to the user, and then update the previous or old ciphertext.

1.3 Organization

The remainder of the paper is structured as follows. In Section 2, we first recall some background knowledge used in this paper. In Section 3, the formal definition and security model of AR-CP-ABE are given. The detailed construction of the proposed scheme is described in Section 4 and the security proof in Section 5. In Section 6, we compare our work with the other related works on functionalities and efficiency. Finally, in Section 7, a conclusion and the future work are given.

2 Preliminaries

2.1 Linear Secret Sharing Scheme

Suppose that $\mathcal{L} = \{\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_n\}$ is an attribute name universe, and for $\forall \mathcal{L}_i \in \mathcal{L}$, the set of attribute value is

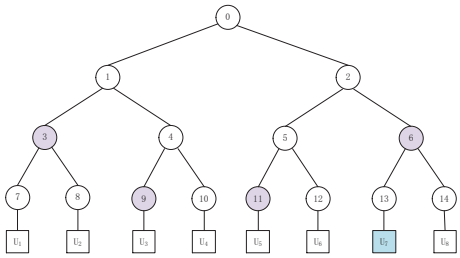


Figure 1: Binary tree $\mathcal{T}$

$\mathcal{L}_i = \{a_{i,1}, a_{i,2}, \cdots a_{i,n_i}\}$. A linear secret-sharing scheme (LSSS) [17] can stand for an access control policy by (M, ρ), where M is an $l \times n$ matrix and ρ is a mapping from the rows of M to attribute names in $\mathcal{L}$. The LSSS comprises of two algorithm:

1) **Share** s**.** The purpose of the algorithm is to hide a value $s \in \mathbb{Z}_p$. Choose a vector $\vec{v} = (s, v_2, \cdots, v_n)^\top$, where $v_2, \cdots, v_n \in \mathbb{Z}_p$ are chosen randomly. Compute $\lambda_i = M_i \cdot \vec{v}$ as a sharing of s, where λ_i matches with the attribute name $\rho(i)$, and M_i is the i-th row vector of M.

2) **Reconstruct** s**.** The algorithm is utilized to recover s. Suppose that S is an authorized set, where S satisfies the access policy (M, ρ) and $I = \{i : \rho(i) \in I_S\} \subseteq \{1, 2, \cdots, l\}$. Then, some coefficients $\{c_i | i \in I\}$ such that $\sum_{i\in I} c_i\lambda_i = s$ will be found.

Set $\mathcal{S} = (I_S, S)$ as a user attribute set, where $I_S \subseteq \mathcal{L}$ is a set of user attribute name, and $S = \{s_i\}_{i\in I_S}$ stands for attribute values. Furthermore, the access policy is represented by $W = (M, \rho, T)$, where M is an $l\times n$ matrix, ρ is a mapping from rows of M to attribute names in $\mathcal{L}$ that each attribute name can occur only once, and $T = \{t_{\rho(i)}\}_{i\in[1,l]}$ is the attribute value related to (M, ρ). Let $\mathcal{S} \in W$ denote $\mathcal{S}$ matches W, which means that there exists a set $I = \{i : \rho(i) \in I_S\} \subseteq \{1, 2, \cdots, l\}$ satisfying W and for $\forall i \in I$, $s_{\rho(i)} = t_{\rho(i)}$, and $\mathcal{S} \notin W$ denote $\mathcal{S}$ dose not match W. Finally, the access policy removing the attribute values is represented by $\overline{W} = (M, \rho)$.

2.2 Binary Tree

A set of all users in the system and a revocation list are represented by $\mathcal{U}$ and $\mathcal{R}$, respectively. A binary tree $\mathcal{T}$ [20] is described as:

- Each leaf node is related to a user U. Set $|\mathcal{U}|$ as the total number of users. Supposed that the nodes are numbered by natural numbers. Concretely, 0 is the serial number of the root node and $2|\mathcal{U}| - 2$ is the last.

- Let $path(i)$ be a path from the root node to the node related to i and $cover(\mathcal{R})$ be a minimum set of the nodes that can cover all users that are not in $\mathcal{R}$, we call it the minimum cover set.

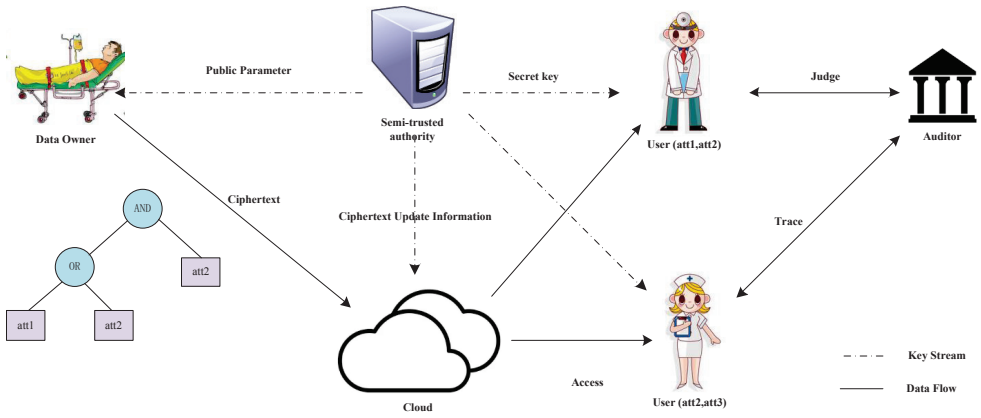


Figure 2: System construction of AR-CP-ABE

As depicted in Figure 1, given a revocation list $\mathcal{R} = \{U_4, U_6\} = \{10, 12\}$, then $cover(\mathcal{R}) = \{3, 9, 11, 6\}$. From the tree, the path of U_7 : $path(U_7) = path(13) = \{0, 2, 6, 13\}$ can be obtained. Thus, the intersection $j = cover(\mathcal{R}) \cap path(U_7) = \{6\}$ can be computed.

2.3 Hardness Assumptions

Now we review three well-known complexity assumptions [21–23] that the security of the proposed scheme are reducible to.

Definition 1. *Let $\mathbb{G}$ be a multiplication cyclic groups of prime order p, and g be a generator of $\mathbb{G}$. Given a tuple (g, g^z), where $z \in \mathbb{Z}_p^*$, the Discrete Logarithm Problem (DLP) is to output z. Furthermore, the DLP hardness assumption holds if no PPT adversary $\mathcal{A}$ can calculate z with non-negligible advantage.*

Definition 2. *Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplication cyclic groups of prime order p, g be a generator of $\mathbb{G}$, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ be a bilinear map. Given $Y = (g, g^s, g^d, g^{d^2}, \cdots, g^{d^q}, g^{d^{q+2}}, \cdots, g^{d^{2q}})$, where $s, d \in \mathbb{Z}_p^*$, the q-Bilinear Diffie-Hellman Exponent (q-BDHE) problem is to distinguish $e(g, g)^{d^{q+1} \cdot s}$ from an element Z that is selected in $\mathbb{G}_T$ randomly. Moreover, the q-BDHE hardness assumption holds if no PPT adversary $\mathcal{A}$ can solve the q-BDHE problem with non-negligible advantage.*

Definition 3. *Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplication cyclic groups of prime order p, g be a generator of $\mathbb{G}$, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ be a bilinear map. Given $(l+1)$-tuple $(g, g^x, g^{x^2}, \cdots, g^{x^l})$, where $x \in \mathbb{Z}_p^*$, the l-Strong Diffie-Hellman (l-SDH) problem is to output a tuple $(c, g^{1/(a+c)})$. Furthermore, the l-SDH hardness assumption holds if no PPT adversary $\mathcal{A}$ can calculate $(c, g^{1/(a+c)})$ with non-negligible advantage.*

3 System and Security Models

In this section, the system architecture, the formal definition, and a series of security models about AR-CP-ABE will be given.

3.1 System Framework

There are five entities in the system framework of AR-CP-ABE, as depicted in Figure 2.

- **Semi-trusted authority**. A semi-trusted authority can setup the system, publish the public parameters, and generate the decryption keys and the update keys for the users and cloud server, respectively.
- **Data owner (Patient)**. The data owners can encrypt the health record according to the specified access policy to the cloud.
- **Cloud server**. A cloud, which is honest-but-curious, can store the ciphertext for the data owner and update the ciphertext by using of the update key from the authority.
- **User (Medical personnel and specialists, etc)**. A user can decrypt the ciphertext successfully when the user's attributes can satisfy the access policy and identity is not in the revocation list.
- **Auditor**. A trusted auditor is responsible for the audit and revocation procedure, and returns the corresponding results to the users.

3.2 The Formal Definition of AR-CP-ABE

A formal AR-CP-ABE scheme mainly includes seven algorithms as follows:

- **Setup**$(\lambda, \mathcal{T}, \mathcal{L}) \to (PP, MSK)$. The algorithm is executed by the authority. Take as input the security parameter λ, a binary tree $\mathcal{T}$ and the attribute universe $\mathcal{L}$, then output the public parameters PP and the master secret key MSK that is kept secretly.

- **KeyGen**$(MSK, U, \mathcal{S}) \rightarrow SK$. The authority interacts with a user U, and runs the algorithm. The authority inputs the master secret key MSK, the user U and its attribute set $\mathcal{S} = (I_S, S)$, then sends the intermediate key for the user.
- **Encryption**$(PP, m, (M, \rho, T), \mathcal{R}) \rightarrow CT$. The algorithm is executed by the data owner that inputs the public parameter PP, a message m, the newly revocation list $\mathcal{R}$, and the access policy $W = (M, \rho, T)$, and generates a ciphertext CT.
- **Decryption**$(SK, CT) \rightarrow m$. After inputting the ciphertxt CT and her or his decryption key SK, the ciphertext can be decrypted successfully when the user's attributes can satisfy the access policy and identity is not in $\mathcal{R}$.
- **KeySanityCheck**$(SK) \rightarrow 1/\perp$. As a third party, the auditor runs this algorithm to evaluate whether the key is formal well.
- **Trace**$(SK_{suspected}, PP, \mathcal{R}) \rightarrow U/authority/\perp$. The auditor executes this algorithm and outputs who is a dishonest party, and manages the revocation list $\mathcal{R}$.
- **CTUpdate**$(CT, R', X') \rightarrow CT'$. The ciphertext update algorithm is executed by the cloud. Take the ciphertext CT, the new revocation list $\mathcal{R}'$ and the update key X' from the authority as input, and output an updated ciphertext CT'.

3.3 IND-CPA Security Model

The IND-CPA security [17] of AR-CP-ABE scheme is depicted by a game executed between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$. Specific steps are as follow:

- **Initialization**: $\mathcal{A}$ determines a challenged access policy $W^* = (M^*, \rho^*, T^*)$ and a revocation list $\mathcal{R}^*$, where M^* is an $l^* \times n^*$ matrix with $n^* \leq q$, ρ is a mapping from rows of M to attribute names in $\mathcal{L}$, and $T^* = \{t_{\rho^*(i)}\}_{i \in [1, l^*]}$ is the attribute value related to (M^*, ρ^*).
- **Setup**: The challenger $\mathcal{C}$ generates a master secret key MSK and public parameters PP, and submits PP to $\mathcal{A}$ by utilizing the Setup algorithm.
- **Phase 1**: In this phase, the adversary $\mathcal{A}$ submits a series of user attribute sets $\{(U_1, \mathcal{S}_1), \cdots, (U_q, \mathcal{S}_q)\}$ to $\mathcal{C}$.
  - **Case 1**: If $\mathcal{S}_i \in W^*$ and $U \notin \mathcal{R}^*$, then $\mathcal{C}$ aborts.
  - **Case 2**: If $\mathcal{S}_i \notin W^*$ or $U \in \mathcal{R}^*$, $\mathcal{C}$ generates the intermediate keys for $\mathcal{A}$ by running the KeyGen algorithm.
- **Challenge**: $\mathcal{A}$ chooses two equal-length messages m_0, m_1 and sends them to $\mathcal{C}$. Then $\mathcal{C}$ flips a coin $\upsilon \in \{0, 1\}$ randomly and encrypts m_υ under the access policy (M^*, ρ^*) and the revocation list $\mathcal{R}^*$. Finally, the ciphertext CT^* will be sent to $\mathcal{A}$ by $\mathcal{C}$.
- **Phase 2**: Phase 2 is as same as Phase 1.
- **Guess**: $\mathcal{A}$ outputs a guess υ' of υ. $\mathcal{A}$ will win the game if $\upsilon' = \upsilon$.

The advantage of $\mathcal{A}$ that wins the above game is defined as

$$Adv(\mathcal{A}) = |\Pr[\upsilon' = \upsilon] - 1/2|.$$

Definition 4. *The AR-CP-ABE scheme is IND-CPA secure if all the PPT adversaries have at most negligible advantage in the above game.*

3.4 Accountability Security Model

The accountability security model that used by Ning *et al.* [11] includes three security games: *Dishonest-Authority* game, *Dishonest-User-I* game and *Dishonest-User-II* game. We also use the accountability security model in the proposed scheme.

1) ***Dishonest-Authority game***. The meaning of the game is that the adversarial authority attempts to forge user's key family number ω in the user's decryption key. The *Dishonest-Authority* game for the proposed scheme proceeds as follows.
   - **Setup**: The adversary $\mathcal{A}$ (dishonest authority) submits the public parameters PP to $\mathcal{C}$ by calling the Setup algorithm.
   - **Key Generation**: $\mathcal{A}$ invokes the KeyGen algorithm to generate a intermediate key for $\mathcal{C}$. $\mathcal{C}$ can abort the game when intermediate key is not well-formed.
   - **Key Forgery**: $\mathcal{A}$ outputs a forged decryption key SK' associated with U. Then $\mathcal{C}$ checks whether SK' is well-formed. The $\mathcal{C}$ can abort the game if SK' is not well-formed.

   Suppose that the event that the adversary wins the game is represented by ζ. The advantage of the adversary in *Dishonest-Authority* game is defined as

   $$Adv(\mathcal{A}) = \Pr[\zeta].$$

2) ***Dishonest-User-I game***. The intuition under the game is that a decryption key of a new user U cannot be forged by an adversarial user. The *Dishonest-User-I* game will be carried out as follows.
   - **Setup:** The challenger $\mathcal{C}$ generates a master key MSK and submits the public parameters PP by calling the Setup algorithm.
   - **Key Query:** $\mathcal{A}$ submits the attribute sets $\{(U_1, \mathcal{S}_1), \cdots, (U_q, \mathcal{S}_q)\}$ to $\mathcal{C}$ for requesting the intermediate keys. Then $\mathcal{C}$ invokes the KeyGen algorithm to generate the intermediate keys.
   - **Key Forgery:** $\mathcal{A}$ outputs a forged key SK^*. If $Trace(SK^*, \mathcal{R}, PP) \neq \perp$ and $Trace(SK^*, \mathcal{R}, PP) \notin \{U_1, \cdots, U_q\}$, $\mathcal{A}$ wins the game.

The advantage of $\mathcal{A}$ in the above game is defined as

$$Adv(\mathcal{A}) = \Pr[Trace(SK^*, \mathcal{R}, PP) \neq \perp \cup Trace(SK^*, \mathcal{R}, PP) \notin \{U_1, \cdots, U_q\}].$$

3) ***Dishonest-User-II game***. The meaning of the game is that another key family number (represented by ω) cannot be forged by an adversarial user. The *Dishonest-User-II* game for the proposed scheme will be carried out as follows.

- **Setup:** The challenger $\mathcal{C}$ generates a master secret key MSK and the public parameters PP by executing the Setup algorithm. Then $\mathcal{C}$ submits PP to $\mathcal{A}$.
- **Key Query:** $\mathcal{A}$ submits attribute sets $\{(U_1, \mathcal{S}_1), \cdots, (U_q, \mathcal{S}_q)\}$ to $\mathcal{C}$. Then $\mathcal{C}$ generates the intermediate keys for $\mathcal{A}$ by calling the KeyGen algorithm.
- **Key Forgery:** $\mathcal{A}$ outputs a forged key SK^* of the user U. $\mathcal{A}$ wins the game if $(U, c) = (U_i, c_i) \in \{(U_1, c_1), \cdots, (U_q, c_q)\}, \omega \neq \omega_i$ and SK is well-formed.

The advantage of $\mathcal{A}$ in the above game is defined as

$$Adv(\mathcal{A}) = \Pr[Trace(SK^*, \mathcal{R}, PP) \in \{U_1, \cdots, U_q\} \cup Audit(SK^*) \rightarrow innocent].$$

Definition 5. *The AR-CP-ABE scheme is accountable if all the PPT adversaries have at most negligible advantage in the above three games.*

3.5 Collusion Resistance Security Model

In order to apply to the proposed scheme with user revocation, we modify the security model of Li *et al.* [14] by replacing the attribute with the user. The game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ is defined as follows:

- **Initialization**: $\mathcal{A}$ sends a challenged access policy $W^* = (M^*, \rho^*, T^*)$ and a revocation list $\mathcal{R}^*$ to $\mathcal{C}$.
- **Setup**: The challenger $\mathcal{C}$ generates a master secret key MSK and public parameters PP, and submits PP to $\mathcal{A}$ by calling the Setup algorithm.
- **Phase 1**: In this phase, the adversary $\mathcal{A}$ can issue two types of queries as follows:
  - **Type-I key query** $\langle U_I, \mathcal{S}_I \rangle$: User attribute set $\mathcal{S}_I \notin W^*$, but the user U_I is unrevoked. $\mathcal{C}$ interacts with $\mathcal{A}$ and then generates a decryption key by running the KeyGen algorithm. Finally, $\mathcal{C}$ returns the key to $\mathcal{A}$.
  - **Type-II key query** $\langle U_{II}, \mathcal{S}_{II} \rangle$: User U_{II} already has been revoked, while the user attribute set $\mathcal{S}_{II} \in W^*$. $\mathcal{C}$ interacts with $\mathcal{A}$ and then generates a decryption key by running the KeyGen algorithm. Later, $\mathcal{C}$ sends the key to $\mathcal{A}$.
- **Challenge**: After receiving two equal-length messages m_0, m_1 from $\mathcal{A}$, $\mathcal{C}$ flips a coin $b \in \{0, 1\}$ randomly and encrypts m_b by using the challenged access policy W^* and the revocation list $\mathcal{R}^*$. Finally, the ciphertext CT^* will be sent to $\mathcal{A}$.
- **Phase 2**: Phase 2 is as same as Phase 1.
- **Guess**: $\mathcal{A}$ outputs a guess b' of b. $\mathcal{A}$ will win the game if $b' = b$.

The advantage of $\mathcal{A}$ that wins the above game is defined as

$$Adv(\mathcal{A}) = |\Pr[b' = b] - 1/2|.$$

Definition 6. *The AR-CP-ABE scheme with user revocation is secure against collusion attacks if all the PPT adversaries have at most negligible advantage in the above game.*

4 Construction of AR-CP-ABE

Inspired with Han *et al.*'s scheme [17] and Li *et al.*'s scheme [11], we will construct an AR-CP-ABE scheme in this section.

- **Setup**$(\lambda, \mathcal{L}, \mathcal{T}) \rightarrow (PP, MSK)$. The algorithm is executed by the authority, and takes as input a security parameter λ, an attribute universe $\mathcal{L}$ and a binary tree $\mathcal{T}$ associated with user U that $U \in \mathcal{U}$. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplication cyclic groups of prime order p, g be a generator of $\mathbb{G}$, and $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be a bilinear mapping. The algorithm is implemented as follows:
  1) Pick $h, u \in \mathbb{G}$ and $a, \alpha \in \mathbb{Z}_p$ randomly.
  2) For each node of $\mathcal{T}$, randomly select $\{x_i\}_{i=0}^{2|\mathcal{U}|-2} \in \mathbb{Z}_p^*$, and compute $\{y_i = g^{x_i}\}_{i=0}^{2|\mathcal{U}|-2}$.
  3) Select a probabilistic symmetric encryption scheme (Enc, Dec) from $\{0,1\}^*$ to $\mathbb{Z}_p$, which sets $\bar{k} \in \mathbb{Z}_p$ as secret key. Then, the authority sends $\bar{k}$ to the auditor.

  Finally, the public parameters are published as:

  $$PP = (p, \mathbb{G}, \mathbb{G}_T, e, g, h, u, e(g,g)^{\alpha}, g^a, \{y_i\}_{i=0}^{2|\mathcal{U}|-2}),$$

  and the master key is kept secretly as:

  $$MSK = (a, \alpha, \{x_i\}_{i=0}^{2|\mathcal{U}|-2}, \bar{k}).$$

- **KeyGen**$(MSK, U, \mathcal{S}) \rightarrow (SK)$. The authority interacts with a user U whose attribute set is $\mathcal{S} = (I_S, S)$, where I_S is a set of user attribute name, and $S = \{s_i\}_{i \in I_S}$ stands for a set of attribute values. Then, the authority computes $c = Enc_{\bar{k}}(i_d)$, where i_d is the value of the leaf node about the user U. The algorithm is executed as follows:

1) The user randomly chooses $\omega \in \mathbb{Z}_p^*$, computes $H = h^{\omega}$, and sends H to the authority. The user also needs to make a proof of knowledge to the authority with regard to the discrete logarithm of H.
2) If the proof of knowledge is valid, the authority selects $r \in \mathbb{Z}_p$, and for $\forall \tau \in I_S$, computes as follows: $(K' = c, K = g^{\frac{\alpha}{a+c}} \cdot H^r, L = g^r, L' = g^{a \cdot r}, L'' = g^{r \cdot x_{i_d}}, \{K_\tau = g^{x_{i_d} \cdot s_\tau \cdot r} \cdot u^{-(a+c) \cdot r}\}_{\tau \in I_S})$.
3) Suppose $path(i_d) = \{i_0, \cdots, i_d\}$, where $i_0 = root$ and i_d is the value of a leaf node about the user U in the tree. The authority computes $\{KU_i = g^{r \cdot \frac{x_{i_d}}{x_i}}\}_{i \in path(i_d)}$ for the user U, then sends a tuple (U, i_d) and the intermediate key $(K', K, L, L', L'', \{KU_i\}_{i \in path(i_d)}, \{K_\tau\}_{\tau \in I_S}, \mathcal{S})$ to the auditor and the user, respectively.
4) Finally, the user sets the full decryption key $SK = (K', K, T' = \omega, L, L', L'', \{K_\tau\}_{\tau \in I_S}, \{KU_i\}_{i \in path(i_d)}, \mathcal{S})$, and the auditor adds the tuple (U, i_d) in the list LN that is used to trace.

- **Encryption**$(PP, m, (M, \rho, T), \mathcal{R}) \to CT$. Taking as input the public parameters PP, a message $m \in \mathbb{G}_T$, the latest revocation list $\mathcal{R}$, and an access policy $W = (M, \rho, T)$, where $T = \{t_{\rho(i)}\}_{i \in [1,l]}$ is the attribute value, a data owner runs the algorithm in the following.

1) Choose a vector $\vec{v} = (s, v_2, \cdots, v_n)^\top$ randomly, where $s, v_2, \cdots, v_n \in \mathbb{Z}_p$, and calculate $\vec{\lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_l)^\top = M\vec{v}$.
2) Select $k_i \in \mathbb{Z}_p$ randomly, where $i \in [1, l]$, and compute a partial ciphertext based on the access policy W: $(C = m \cdot e(g,g)^{\alpha s}, C_0 = g^s, C_0' = g^{a \cdot s}, \{C_{i,1} = h^{\lambda_i} \cdot u^{k_i}, C_{i,2} = g^{-k_i \cdot t_{\rho(i)} + \lambda_i}, C_{i,3} = g^{k_i}\}_{i \in [1,l]})$.
3) Compute a partial ciphertext related to the revocation list $\mathcal{R}$: $(\{T_j = y_j^s\}_{j \in cover(\mathcal{R})})$.
4) Finally, send a full ciphertext as follows: $CT = (C, C_0, C_0', \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i \in [1,l]}, \{T_j\}_{j \in cover(\mathcal{R})}, \mathcal{R}, \overline{W})$, where $\overline{W} = (M, \rho)$ is the access policy that removes the attribute value set.

- **Decryption** $(SK, CT) \to m$. Taking the ciphertxt CT as input, the user who owns the full decryption key $SK = (\mathcal{S}, K', K, T', L, L', L'', \{KU_i\}_{i \in path(i_d)}, \{K_\tau\}_{\tau \in I_S})$ and the attribute set $\mathcal{S}$ can implement the following algorithm.

1) For $U \notin \mathcal{R}$, there must exist a node j that $j \in cover(\mathcal{R}) \cap path(U)$. Suppose that $path(U) = \{i_0, \cdots, i_{dep(j)}, \cdots, i_d\}$, where $i_{dep(j)} = j$, and compute

$$B = e(KU_j, T_j)^{T'} = e(g,g)^{r \cdot x_{i_d} \cdot s \cdot \omega}.$$

2) Let $I = \{i : \rho(i) \in I_S\} \subseteq [1, 2, \cdots, l]$. There exist coefficients $\{c_i | i \in I\}$ such that $\sum_{i \in I} c_i \lambda_i = s$. And then compute

$$\begin{aligned} E =& [e(L^{K'} \cdot L', C_{i,1}) \cdot e(L'', C_{i,2}) \cdot e(K_{\rho(i)}, C_{i,3})]^{T'} \\ =& e(g,h)^{(a+c) \cdot r \cdot \lambda_i \cdot \omega} \cdot e(g,g)^{r \cdot x_{i_d} \cdot \lambda_i \cdot \omega}, \\ F =& \prod_{i \in I} (E)^{c_i} = e(g,h)^{(a+c) \cdot r \cdot s \cdot \omega} \cdot e(g,g)^{r \cdot x_{i_d} \cdot s \cdot \omega}, \\ D =& e(K, C_0^{K'} \cdot C_0') = e(g^{\frac{\alpha}{a+c}} \cdot H^r, g^{(a+c) \cdot s}) \\ =& e(g,g)^{\alpha \cdot s} \cdot e(g,h)^{(a+c) \cdot r \cdot s \cdot \omega}. \end{aligned}$$

3) Finally, recover the message as $m = \frac{C \cdot F}{D \cdot B}$.

- **KeySanityCheck** $(SK) \to (1/\perp)$. The algorithm is used to check whether a decryption key $SK = (K', K, T', L, L', L'', \{KU_i\}_{i \in path(i_d)}, \{K_\tau\}_{\tau \in I_S}, \mathcal{S})$ is well-formed. The auditor executes this algorithm as follows:

$$T', K' \in \mathbb{Z}_p, K, L, L', L'', K_\tau \in \mathbb{G}, \tag{1}$$

$$e(g, L') = e(g^a, L) \neq 1, \tag{2}$$

$$e(K, g^a \cdot g^{K'}) = e(g,g)^{\alpha} \cdot e(L^{K'} \cdot L', h^{T'}) \neq 1, \tag{3}$$

$$\exists \tau \in I_S, \text{s.t. } e(K_\tau, g) \cdot e(L \cdot L', u) = e(L'', g)^{s_\tau} \neq 1. \tag{4}$$

The decryption key SK is reviewed as a well-formed key only if it satisfies these Equations (1,2,3,4), and then the algorithm outputs 1; otherwise $\perp$.

- **Trace**$(SK_{suspected}, PP, \mathcal{R}) \to (U/authority/\perp)$. The auditor runs this algorithm. If the decryption key $SK_{suspected}$ is not well-formed, then the algorithm outputs $\perp$. Otherwise, firstly obtain $i_d = Dec_{\bar{k}}(K')$ and $T' = \omega$ from $SK_{suspected}$, and then search i_d in $LN = \{U, i_d\}$. If there not exists the same value in LN, the algorithm outputs the authority, which means that the dishonest authority fakes a user. Otherwise, the algorithm compares $T' = \omega$ in $SK_{suspected}$ with ω_U associated with the real user U. If $\omega \neq \omega_U$, the algorithm outputs the authority as a dishonest party and claim that the user is innocent. If $\omega = \omega_U$, the algorithm outputs the user U, which indicates the user U is dishonest, and generates the new revocation list $\mathcal{R}' = \mathcal{R} \bigcup \{U\}$.

- **CTUpdate**$(CT, \mathcal{R}', X') \to (CT')$. The ciphertext updated algorithm is executed by the cloud. The authority selects $\eta \in \mathbb{Z}_p$ randomly, calculates $X' = \{x_i' = \eta \cdot x_i \mod p\}_{i=0}^{2|\mathcal{U}|-2}$, and then sends them to the cloud. The update key X', the latest revocation list $\mathcal{R}'$ and the ciphertext CT are put into the algorithm by the cloud. Subsequently, the algorithm will output the new ciphertext CT' associated with the new revocation list $\mathcal{R}'$. For $j' \in cover(\mathcal{R}')$, there are two case:

1) If there exists $j \in cover(\mathcal{R})$ such that $j = j'$, then set $T_{j'} = T_j$.
2) If there exists $j \in cover(\mathcal{R})$ such that j is an ancestor of j', suppose that

$$path(j') = path(j) \bigcup \{i_{dep(j)+1}, \cdots i_{dep(j')}\},$$

where $i_{dep(j)} = j$ and $i_{dep(j')} = j'$. Let $Y_j = T_j$, compute iteratively

$$Y_{i_{k+1}} = (Y_{i_k})^{\frac{x'_{i_{k+1}}}{x'_{i_k}}} = y^s_{i_{k+1}},$$

where $k = dep(j), \cdots, dep(j') - 1$ and set $T_{j'} = Y_{j'}$. The other partial ciphertext remains to be unchanged, and then the updated ciphertext is $CT = (C, C_0, C'_0, \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i\in[1,l]}, \{T_{j'}\}_{j'\in cover(\mathcal{R}')}, \mathcal{R}', \overline{W})$.

5 Security Analysis

In this section, we first prove the IND-CPA security and accountability, and then give the proof of resistance against the collusion attacks between a revoked user and an unrevoked user.

5.1 IND-CPA Security

In the proposed AR-CP-ABE scheme, we only prove the security of the fresh ciphertext, since the distribution of updated ciphertext is as same as the fresh ciphertext. The security proof of our AR-CP-ABE scheme will be described below.

Theorem 1. *If the decisional q-BDHE hardness assumption holds, there is no polynomial time adversary that can break our AR-CP-ABE scheme with non-negligible advantage under the selective access policy and chosen plaintext attacks.*

Proof. Suppose that there exists a PPT adversary $\mathcal{A}$ that can break our scheme with a non-negligible advantage ε, then we can construct a challenger $\mathcal{C}$ that can solve the q-BDHE problem with the advantage $\varepsilon/2$.

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplication cyclic groups of prime order p, g be a generator of $\mathbb{G}$, and the mapping $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ be a bilinear map. Suppose that $q > 2|\mathcal{U}| - 2$. Then $\mathcal{C}$ randomly flips a fair coin $\mu = \{0, 1\}$. Given $Y' = (g, g^s, g^d, g^{d^2}, \cdots, g^{d^q}, g^{d^{q+2}}, \cdots, g^{d^{2q}})$, $\mathcal{C}$ sets $Z = e(g, g)^{d^{q+1}s}$, if $\mu = 0$; Otherwise, $\mathcal{C}$ selects $Z \in \mathbb{G}_T$ randomly. Furthermore, in order to utilize $\mathcal{A}$ to distinguish Z, $\mathcal{C}$ should simulate a challenger for $\mathcal{A}$. Thus, the simulation is as follows:

- **Initialization:** $\mathcal{A}$ chooses a challenge access policy $W^* = (M^*, \rho^*, T)$ and a revocation list $\mathcal{R}^*$, where M^* is an $l^* \times n^*$ matrix and $n^* \le q$, ρ^* is a mapping from rows of M^* to the attribute name, and $T = \{t_{\rho^*(i)}\}_{i\in[1,l^*]}$ is the attribute value related to (M^*, ρ^*).
- **Setup:** $\mathcal{C}$ generates the public parameter as follows:
  1) Select $\alpha' \in \mathbb{Z}_p$ and set $e(g, g)^{\alpha} = e(g^d, g^{d^q}) \cdot e(g, g)^{\alpha'}$, which means implicitly $\alpha = \alpha' + d^{q+1}$. Then pick $a \in \mathbb{Z}_p$, compute g^a, and set $h = g^d, u = g^{d^q}$.
  2) Given the revocation list $\mathcal{R}^*$, let $I_{\mathcal{R}^*} = \{i \in path(U)|U \in \mathcal{R}^*\}$, and select $v_i \in \mathbb{Z}_p, \forall i = 0, 1, \cdots, 2|\mathcal{U}| - 2$. If $i \in I_{\mathcal{R}^*}$, set $y_i = g^{v_i} g^{d^i}$, which implies $x_i = v_i + d^i$. Otherwise, let $y_i = g^{v_i} g^{d^q}$, which means implicitly $x_i = v_i + d^q$.

  The public parameters are published as follows:

$$PP = (p, \mathbb{G}, \mathbb{G}_T, e, g, h, u, e(g, g)^{\alpha}, g^a, \{y_i\}_{i=0}^{2|\mathcal{U}|-2}).$$

- **Phase 1:** To request the related intermediate keys, $\mathcal{A}$ picks randomly ω, computes $H = h^{\omega}$ with a zero-knowledge proof, and submits H and a series of user attribute sets $\{U, \mathcal{S} = (I_S, S)\}$ to $\mathcal{C}$, where I_S and $S = \{s_i\}_{i\in I_S}$ are the attribute name and attribute value of the user, respectively. Similar to the case in A-IBE [27], utilizing the knowledge extractor, $\mathcal{C}$ can extract ω. Then for each attribute value $s_\tau \in S$ and $i \in \{1, 2, \ldots, l^*\}$, if $s_\tau = t_{\rho^*(i)}$, set $u_\tau = s_\tau + \sum_{n=1}^{n^*} d^n M^*_{k,n}$; Otherwise, let $u_\tau = s_\tau$. According to the four combinations that whether the attribute satisfies the access policy and whether the user is revoked, $\mathcal{C}$ runs as follows:
  - **Case 1:** If $\mathcal{S} \in W^*$ and $U \notin \mathcal{R}^*$, then $\mathcal{C}$ aborts.
  - **Case 2:** If $\mathcal{S} \in W^*$ and $U \in \mathcal{R}^*$, $\mathcal{C}$ executs as follows:
    1) Choose $c \in \mathbb{Z}_p$ randomly, set $K' = c$, and compute $K, L, L', L'', \{K_\tau\}_{\tau\in I_S}$ in the followings:

$$\begin{aligned}
K &= g^{\frac{\alpha'}{a+c}} \left(g^{\frac{d^q}{a+c}}\right)^{\frac{M^*_{i,1}}{M^*_{i,2}}} = g^{\frac{\alpha}{a+c}} h^{r\omega},\\
L &= [(g^{d^q})^{\frac{1}{(a+c)\omega}}]^{-1} [(g^{d^{q-1}})^{\frac{1}{(a+c)\omega}}]^{\frac{M^*_{i,1}}{M^*_{i,2}}} = g^r,\\
L' &= (g^r)^a, \qquad L'' = g^{r\cdot(v_{i_d} + d^{i_d})} = g^{r x_{i_d}},\\
K_\tau &= [(g^{(v_{i_d} + d^{i_d}) d^q})^{\frac{s_\tau}{(a+c)\omega}}]^{-1}\\
&\quad \cdot [(g^{(v_{i_d} + d^{i_d}) d^{q-1}})^{\frac{s_\tau}{(a+c)\omega}}]^{\frac{M^*_{i,1}}{M^*_{i,2}}}\\
&\quad \cdot (g^{d^{2q}})^{\frac{1}{\omega}} [(g^{d^{2q-1}})^{\frac{M^*_{i,1}}{M^*_{i,2}}}]^{-\frac{1}{\omega}}\\
&\quad \cdot \Big[(\prod_{k=2}^{n^*} g^{d^{q+k} M^*_{i,k}})^{-1}\\
&\quad \cdot (\prod_{k=1,k\neq 2}^{n^*} g^{d^{q+k-1} M^*_{i,k}})^{\frac{M^*_{i,1}}{M^*_{i,2}}}\Big]^{\frac{1}{(a+c)\omega}}\\
&= g^{x_{i_d} u_\tau r} u^{-(a+c)r},
\end{aligned}$$

which means implicity $r = -\frac{d^q}{\omega(a+c)} + \frac{d^{q-1}}{\omega(a+c)} \cdot \frac{M^*_{i,1}}{M^*_{i,2}}$.

2) Suppose that $path(i_d) = \{i_0, \cdots, i_d\}$, where $i_0 = root$ and i_d is the value of the leaf node related to the user U. Since $U \in \mathcal{R}^*$, then $i_d \in I_{\mathcal{R}^*}$ and $x_{i_d} = v_{i_d} + d^{i_d}$. For $i \in path(i_d)$, $\mathcal{C}$ can compute

$$KU_i = \Big[(g^{d^q})^{-1} \cdot (g^{d^{q-1}})^{\frac{M^*_{i,1}}{M^*_{i,2}}}\Big]^{\frac{(v_{i_d}+d^{i_d})}{(v_i+d^i)(a+c)\omega}} = g^{r\frac{x_{i_d}}{x_i}}.$$

– **Case 3:** If $\mathcal{S} \notin W^*$ and $U \in \mathcal{R}^*$, $\mathcal{C}$ does as follows:

1) Select $\vec{\omega} = (\omega_1, \cdots, \omega_{n^*}) \in \mathbb{Z}_p^{n^*}$, where $\omega_1 = -1$ and $M^*_i \cdot \vec{\omega} = 0$ for all i such that $\rho^*(i) \in I_S$. Select $c \in \mathbb{Z}_p$ randomly, and set $K' = c$.

2) Choose $t \in \mathbb{Z}_p$ randomly, and calculate K, L, L', L'':

$$K = (g^{\alpha'+dt} \prod_{i=2}^{n^*} g^{\omega_i d^{q+2-i}})^{\frac{1}{a+c}} = g^{\frac{\alpha}{a+c}} h^{r\omega},$$

$$L = [g^{\frac{t}{a+c}} \prod_{i=1}^{n^*} (g^{\omega_i d^{q+1-i}})^{\frac{1}{a+c}}]^{\frac{1}{\omega}} = g^r,$$

$$L' = (g^r)^a, \qquad L'' = g^{r\cdot(v_{i_d}+d^{i_d})} = g^{r x_{i_d}},$$

which means implicity $r = \frac{1}{(a+c)\omega}(t + \omega_1 d^q + \omega_2 d^{q-1} + \cdots + \omega_{n^*} d^{q-n^*+1})$.

3) $\forall \tau \in I_S$, if $\exists i, s.t.\ \rho^*(i) = \tau$ and $s_\tau = t_{\rho^*(i)}$, then $\mathcal{C}$ computes K_τ as follows:

$$K_\tau = \Big[\prod_{j=1}^{n^*} (g^{td^j} \prod_{k=1}^{n^*} g^{\omega_k d^{q+1+j-k}})^{M_{i,j}}\Big]^{\frac{1}{(a+c)\omega}} \cdot \Big(g^{td^q} \prod_{i=1}^{n^*} g^{\omega_i d^{2q+1-i}}\Big)^{-\frac{1}{\omega}} \cdot L^{(v_{i_d}+d^{i_d})s_\tau} = g^{x_{i_d} u_\tau r} u^{-(a+c)r}.$$

Otherwise, $\mathcal{C}$ computes K_τ as follows:

$$K_\tau = L^{(v_{i_d}+d^{i_d})s_\tau} (g^{td^q} \prod_{i=1}^{n^*} g^{\omega_i d^{2q+1-i}})^{-\frac{1}{\omega}}.$$

4) Suppose that $path(i_d) = \{i_0, \cdots, i_d\}$, where $i_0 = root$ and i_d is the value of the leaf node related to the user U. Since $U \in \mathcal{R}^*$, $i_d \in I_{\mathcal{R}^*}$ and $x_{i_d} = v_{i_d} + d^{i_d}$. For $i \in path(i_d)$, $\mathcal{C}$ computes

$$KU_i = \Big(g^t \prod_{i=1}^{n^*} g^{\omega_i d^{q+1-i}}\Big)^{\frac{(v_{i_d}+d^{i_d})}{(v_i+d^i)(a+c)\omega}} = g^{r\frac{x_{i_d}}{x_i}}.$$

– **Case 4:** If $\mathcal{S} \in W^*$ and $U \notin \mathcal{R}^*$, then $K, K', L, L', L'', \{K_\tau\}_{\tau \in I_S}$ can be calculated as

Case 3. Since $U \notin \mathcal{R}^*$, then $i_d \notin I_{\mathcal{R}^*}$ and $x_{i_d} = v_{i_d} + d^q$. Next, for $i \in path(i_d)$, $\mathcal{C}$ sets

$$KU_i = \Big(g^t \prod_{i=1}^{n^*} g^{\omega_i d^{q+1-i}}\Big)^{\frac{(v_{i_d}+d^q)}{(v_i+d^q)(a+c)\omega}} = g^{r\cdot x_{i_d}}.$$

- **Challenge:** $\mathcal{A}$ sends two equal-length messages m_0, m_1 to $\mathcal{C}$. $\mathcal{C}$ computes a challenge ciphertext as follows:

1) $\mathcal{C}$ flips a fair coin $\upsilon \in \{0, 1\}$ and computes $C = m_\upsilon \cdot Z \cdot e(g, g)^{\alpha' s}, C_0 = g^s, C_0' = (g^a)^s$.

2) $\mathcal{C}$ selects $r_2, \cdots, r_{n^*} \in \mathbb{Z}_p^*$ randomly, sets $\vec{v} = (s, sd + r_2, \cdots, sd^{n^*-1} + r_{n^*})^\top \in \mathbb{Z}_p^{n^*}$, and then computes

$$C_{i,1} = \prod_{j=2}^{n^*} (g^{dr_j})^{M^*_{i,j}} \prod_{j=1}^{n^*} (g^{sd^j})^{M^*_{i,j}} g^{-ad^{q+i}},$$

$$C_{i,2} = (g^{t_{\rho^*(i)}})^{-ad^i} \prod_{j=2}^{n^*} (g^{d^j M^*_{i,j}})^{-ad^i} \prod_{j=2}^{n^*} (g^{r_j})^{M^*_{i,j}} \cdot \prod_{j=1}^{n^*} (g^{sd^{j-1}})^{M^*_{i,j}} = g^{-t_i u_{\rho^*(i)} + \lambda_i},$$

$$C_{i,3} = g^{-ad^i}.$$

3) $\forall j \in cover(\mathcal{R}^*)$, since $x_j = v_j + d^q$ and $y_j = g^{v_j + d^q}$, then $\mathcal{C}$ sets $T_j = (g^s)^{v_j + d^q} = y_j^s$.

Finally, $\mathcal{C}$ sends the challenge ciphertext $CT = \big(C, C_0, C_0', \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i \in [1, l^*]}, \{T_j\}_{j \in cover(\mathcal{R}^*)}\big)$ to $\mathcal{A}$.

- **Phase 2:** Phase 2 is the same as Phase 1.

- **Guess:** A guess υ' of υ will be output by $\mathcal{A}$.

1) If $\upsilon = \upsilon'$, $\mathcal{C}$ will output a guess $\mu' = 0$ of μ. In this case, $\mathcal{C}$ sets $Z = e(g, g)^{d^{q+1}}$ and $\mathcal{A}$ will obtain a legal ciphertext. Since the advantage of $\mathcal{A}$ is ε, $|\Pr[\upsilon = \upsilon' | \mu = 0] - \frac{1}{2}| = \varepsilon$. Furthermore, $\Pr[\upsilon = \upsilon' | \mu = 0] = \Pr[\mu = \mu' | \mu = 0]$ can be concluded. Then we have $\Pr[\mu = \mu' | \mu = 0] = \varepsilon + \frac{1}{2}$.

2) If $\upsilon \neq \upsilon'$, $\mathcal{C}$ outputs a guess $\mu' = 1$ of μ. In this case, $\mathcal{C}$ selects $Z \in \mathbb{G}_T$ randomly and $\mathcal{A}$ cannot obtain any information of υ. Thus, the advantage of $\mathcal{A}$ is $\frac{1}{2}$, that is to say, $\Pr[\upsilon \neq \upsilon' | \mu = 1] = \frac{1}{2}$, In addition, $\Pr[\upsilon \neq \upsilon' | \mu = 1] = \Pr[\mu = \mu' | \mu = 1]$ is easily concluded. Therefore, we have $\Pr[\mu = \mu' | \mu = 1] = \frac{1}{2}$.

Finally, the advantage of $\mathcal{C}$ in the game is

$$\begin{aligned}|\Pr[\mu=\mu']-\frac{1}{2}| =& |\Pr[\mu=\mu'|\mu=0]\Pr[\mu=0] \\ &+\Pr[\mu=\mu'|\mu=1]\Pr[\mu=1]-\frac{1}{2}| \\ =&|(\varepsilon+\frac{1}{2})\cdot\frac{1}{2}+\frac{1}{2}\cdot\frac{1}{2}-\frac{1}{2}| \\ =&\frac{1}{2}\varepsilon.\end{aligned}$$

□

5.2 Accountability Security

In this section, we prove the accountability security of AR-CP-ABE scheme by the following three theorems.

Theorem 2. *If the DLP hardness assumption holds, the advantage of an adversary in the Dishonest-Authority game is negligible for AR-CP-ABE scheme.*

Proof. Assume that there exists a PPT adversary $\mathcal{A}$ who has a non-negligible advantage ε in the Dishonest-Authority game, then we can construct a challenger $\mathcal{C}$ that can solve a DLP problem with a non-negligible advantage ε.

Furthermore, in order to utilize $\mathcal{A}$ to solve the DLP problem, $\mathcal{C}$ should interact with $\mathcal{A}$ as follows:

- **Setup:** The adversary $\mathcal{A}$ calls the Setup algorithm and submits the public parameters $PP = (p, \mathbb{G}, \mathbb{G}_T, e, g, h, u, e(g,g)^{\alpha}, g^a, \{y_i\}_{i=0}^{2|\mathcal{U}|-2})$ and $c = E_{\bar{k}}(i_d)$ about a user U to $\mathcal{C}$.
- **Key Query:** $\mathcal{C}$ receives a challenge $H = h^{\omega} \in \mathbb{G}$, where ω is unknown for $\mathcal{C}$. Using rewinding techniques of Zero-knowledge Proof of Knowledge of Discrete log protocol in Goyal's scheme [27], $\mathcal{C}$ can give the required proof without knowledge of ω. Then $\mathcal{A}$ computes $(K', K, L, L', L'', \{KU_i\}_{i\in path(i_d)}, \{K_\tau\}_{\tau\in I_S}, \mathcal{S})$ to $\mathcal{C}$.
- **Key Forgery:** $\mathcal{A}$ outputs a decryption key $SK^* = ((K')^*, K^*, (T')^* = \omega', L^*, (L')^*, (L'')^*, \{K_\tau^*\}_{\tau\in I_S}, \{KU_i^*\}_{i\in path(i_d)}, \mathcal{S})$ associated with U. Then $T' = \omega'$ will be a solution of the discrete logarithm problem if SK^* is well-formed.

If $\mathcal{A}$ can successfully forge a decryption key, $\mathcal{C}$ must solve the discrete logarithm problem. Since DLP hardness assumption cannot be solved in probabilistic polynomial time, there does not exist an $\mathcal{A}$ who has a non-negligible advantage in the *Dishonest-Authority* game. □

Theorem 3. *If the l-SDH hardness assumption holds, the advantage of an adversary in the Dishonest-User-1 game is negligible for the AR-CP-ABE scheme under $q < l$, where q is the number of key query.*

Proof. Suppose that there exists a PPT adversary $\mathcal{A}$ who has a non-negligible advantage ε in the *Dishonest-User-1* game with q key queries and $l = q+1$, then we can construct a challenger $\mathcal{C}$ that attacks the l-SDH hardness assumption with a non-negligible advantage ε. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplication cyclic groups of prime order p, g be a generator of $\mathbb{G}$, and $e: \mathbb{G}\times\mathbb{G}\to\mathbb{G}_T$ be a bilinear mapping. Given an l-SDH problem $(g_1, g_1^a, g_1^{a^2}, \cdots, g_1^{a^l})$, where $g_1 \in \mathbb{G}, a \in \mathbb{Z}_p$, the objective of $\mathcal{C}$ is to find a tuple $(c_r, \varpi_r = g_1^{\frac{1}{a+c_r}})$. For $i = 0, 1, \cdots, l$, set $A_i = g_1^{a^i}$. Then the simulation will be executed as follows:

- **Setup:** $\mathcal{C}$ selects randomly q different values $c_1, c_2, \cdots, c_q \in \mathbb{Z}_q^*$ and $\alpha, \theta \in \mathbb{Z}_p, u \in \mathbb{G}$. Let $f(y) = \prod_{i=1}^{q}(y+c_i) = \sum_{i=0}^{q}\alpha_i y^i$, where $\alpha_0, \cdots, \alpha_q \in \mathbb{Z}_p$ are the coefficients of $f(y)$. Then $\mathcal{C}$ executes as follows:
  1) Let $g = \prod_{i=0}^{q}(A_i)^{\alpha_i} = g_1^{f(a)}$, $g^a = \prod_{i=1}^{q+1}(A_i)^{\alpha_{i-1}} = g_1^{f(a)\cdot a}$.
  2) For each node of $\mathcal{T}$, $\mathcal{C}$ chooses $\{x_i\}_{i=0}^{2|\mathcal{U}|-2} \in \mathbb{Z}_p$ randomly, computes $\{y_i = g^{x_i}\}_{i=0}^{2|\mathcal{U}|-2}$ and publishes the public parameters $PP = (p, \mathbb{G}, \mathbb{G}_T, e, g, h = g^{\theta}, u, e(g,g)^{\alpha}, g^a, \{y_i = g^{x_i}\}_{i=0}^{2|\mathcal{U}|-2})$.
- **Key Query:** $\mathcal{A}$ requests q key queries. For i-th query, $\mathcal{A}$ submits $(U_i, \mathcal{S}_i)$ and $H_i = h^{\omega_i}$ to $\mathcal{C}$, where $\omega_i \in \mathbb{Z}_p$. Then set

$$f_i(y) = \frac{f(y)}{y+c_i} = \prod_{j=1, j\neq i}^{q}(y+c_j) = \sum_{j=0, j\neq i}^{q-1}\beta_j y^j,$$

where $\beta_0, \cdots, \beta_{q-1} \in \mathbb{Z}_p$ are the coefficients of $f_i(y)$. $\mathcal{C}$ computes

$$\sigma_i = \prod_{j=0}^{q-1}(A_j)^{\beta_j} = g_1^{f_i(a)} = g_1^{\frac{f(a)}{a+c_i}} = g^{\frac{1}{a+c_i}}.$$

Then $\mathcal{C}$ chooses $r \in \mathbb{Z}_p$ randomly and computes a partial key about (U_i, S_i) as: $(K' = c_i, K = (\sigma_i)^{\alpha} = g^{\frac{\alpha}{a+c_i}}H^r, L = g^r, L' = (g^a)^r, L'' = g^{rx_{i_d}}, \{K_\tau = g^{x_{i_d}s_\tau r}(u^a\cdot u^{c_i})^{-r} = g^{x_{i_d}s_\tau r}u^{-(a+c_i)r}\}_{\tau\in I_S})$. Suppose $path(i_d) = \{i_0, \ldots, i_d\}$, where $i_0 = root$ and i_d is a leaf node the value associated with the user U_i in the tree. $\mathcal{C}$ sets the key component $KU_i = g^{r\cdot\frac{x_{i_d}}{x_i}}$ of the user U_i. Finally, $\mathcal{C}$ sends the intermediate key $(K', K, L, L', L'', \{K_\tau\}_{\tau\in I_S}, \{KU_i\}_{i\in path(i_d)}, \mathcal{S}_i)$ to $\mathcal{A}$.
- **Key Forgery:** $\mathcal{A}$ sends a forged key SK^* to $\mathcal{C}$. Let ξ_1 stand for the event that $\mathcal{A}$ wins the Dishonest-User-1 game. Suppose that SK^* satisfies the conditions of the key sanity check and $K' \notin \{c_1, \cdots, c_q\}$.
  1) If ξ_1 dose not occur, $\mathcal{C}$ chooses a tuple $(c_r, \varpi_r) \in \mathbb{Z}_p \times \mathbb{G}$ as the solution of l-SDH problem.

2) If ξ_1 occurs, $\mathcal{C}$ writes a polynomial $f(y) = \varphi(y)(y+K') + \varphi - 1$, where $\varphi(y) = \sum_{i=0}^{q-1} \varphi_i y^i$ and $\varphi - 1 \in \mathbb{Z}_p^*$. Since $f(y) = \prod_{i=1}^{q}(y+c_i)$, $c_i \in \mathbb{Z}_p^*$ and $K' \notin \{c_1, \cdots, c_q\}$, $(y+K')$ can not divide into $f(y)$. $\mathcal{C}$ sets $\sigma = (K/L^{\theta T'})^{\alpha^{-1}} = g^{\frac{1}{a+K'}} = g_1^{\frac{f(a)}{a+K'}} = g_1^{\varphi(a)} g_1^{\frac{\varphi-1}{a+K'}}$ and then can compute $c_r = K', \varpi_r = (\sigma \cdot \prod_{i=0}^{q-1} A_i^{-\varphi_i})^{\frac{1}{\varphi-1}} = g_1^{\frac{1}{a+K'}}$. Since $e(g_1^a g_1^{c_r}, \varpi_r) = e(g_1^a g_1^{K'}, g_1^{\frac{1}{a+K'}}) = e(g_1, g_1)$, (c_r, ϖ_r) is the solution to the l-SDH problem.

Let ξ_2 denote the event that (c_r, ϖ_r) is the solution to the l-SDH problem. If $\mathcal{C}$ randomly selects (c_r, ϖ_r), ξ_2 occurs with negligible advantage, for simplicity with 0. In the case where $\mathcal{A}$ succeeds and $\gcd(\varphi - 1, p) = 1$, the probability of (c_r, ϖ_r) that satisfies the condition $e(g_1^a g_1^{c_r}, \varpi_r) = e(g_1^a g_1^{K'}, g_1^{\frac{1}{a+K'}})$ is 1. So the probability of $\mathcal{C}$ to solve the l-SDH problem is:

$$\begin{aligned}\Pr[\xi] =& \Pr[\xi|\overline{\mathcal{A} \text{ succeeds}}] \cdot \Pr[\overline{\mathcal{A} \text{ succeeds}}] \\ &+ \Pr[\xi|\mathcal{A} \text{ succeeds} \wedge \gcd(\varphi-1,p) \neq 1] \\ &\quad \cdot \Pr[\mathcal{A} \text{ succeeds} \wedge \gcd(\varphi-1,p) \neq 1] \\ &+ \Pr[\xi|\mathcal{A} \text{ succeeds} \wedge \gcd(\varphi-1,p) = 1] \\ &\quad \cdot \Pr[\mathcal{A} \text{ succeeds} \wedge \gcd(\varphi-1,p) = 1] \\ =& 0 + 0 + 1 \cdot \Pr[\mathcal{A} \text{ succeeds} \wedge \gcd(\varphi-1,p) = 1] \\ =& \varepsilon.\end{aligned}$$

□

Theorem 4. *If the DLP hardness assumption holds, the advantage of an adversary in the Dishonest-User-2 game is negligible for the AR-CP-ABE scheme.*

Proof. Suppose that there exists a PPT adversary $\mathcal{A}$ that has a non-negligible advantage ε in the *Dishonest-User-2* game, then we can construct a challenger $\mathcal{C}$ that can solve the DLP problem (g, g^z) with a non-negligible advantage. In order to utilize $\mathcal{A}$ to obtain z, $\mathcal{C}$ should simulate a challenger for $\mathcal{A}$, and interacts with $\mathcal{A}$ as follows:

- **Setup:** $\mathcal{C}$ sends the public parameters PP to $\mathcal{A}$ by calling the Setup algorithm. Then $\mathcal{C}$ selects $t, \mu \in \mathbb{Z}_p$ randomly and computes $h = g^t, u = g^\mu$.
- **Key Query:** $\mathcal{A}$ submits a series of attribute sets to $\mathcal{C}$ for requesting the intermediate keys. For every query, when $\mathcal{A}$ makes a proof of knowledge of the discrete log of h^ω with respect to h for $\mathcal{C}$, $\mathcal{C}$ will extract the discrete log ω by using a knowledge extractor [27]. Then $\mathcal{C}$ selects $\gamma \in \mathbb{Z}_p$ and computes $(K' = c, K = g^{\frac{\alpha}{a+c}} g^{\omega z \gamma t} = g^{\frac{\alpha}{a+c}} g^{\omega r t}, L = g^{z\gamma} = g^r, L' = g^{az\gamma} = g^{ar}, L'' = g^{x_{i_d} z\gamma} = g^{x_{i_d} r}, \{K_\tau = g^{x_{i_d} s_\tau z \gamma} g^{-(a+c) z\gamma\mu}\}_{\tau \in I_S} = g^{x_{i_d} s_\tau r} g^{-(a+c) r\mu}\}_{\tau \in I_S})$, which means implicitly $r = \gamma \cdot z$. Finally, $\mathcal{C}$ sends $(K', K, L, L', L'', \{K_\tau\}_{\tau \in I_S}, \mathcal{S})$ to $\mathcal{A}$.
- **Key Forgery:** $\mathcal{A}$ outputs a forged key SK^* related with (U, c). Suppose that (U, c) has been queried, let's call that (U_i, c_i), but ω does not equal to ω_i. The key of (U_i, c_i) is $(K' = c, K = g^{\frac{\alpha}{a+c}} h^{\omega r}, L = g^r, L' = g^{ar}, L'' = g^{x_{i_d} r}, \{K_\tau = g^{x_{i_d} s_\tau r} u^{-(a+c)r}\}_{\tau \in I_S}, T' = \omega_i)$. $\mathcal{A}$ outputs a forged key $SK^* = \big((K')^* = c, K^* = g^{\frac{\alpha}{a+c}} h^{\omega^* r^*}, L^* = g^{r^*}, (L')^* = g^{ar^*}, (L'')^* = g^{x_{i_d} r^*}, \{K_\tau^* = g^{x_{i_d} s_\tau r^*} u^{-(a+c) r^*}\}_{\tau \in I_S}, (T')^* = \omega^*\big)$.

Now, we analyze K and K^*, K_τ and K_τ^*. If $\mathcal{A}$ can forge K^* and K_τ^* successfully, then we can suppose that $K^* = K \cdot h^{p_1} \Rightarrow \omega r + p_1 = \omega^* r^*$ and $K_\tau^* = K_\tau^{p_2} \Rightarrow r p_2 = r^*$. Since $\mathcal{A}$ knows $\omega, \omega^*, p_1, p_2$, then $\mathcal{A}$ can get $r = p_1/(\omega^* p_2 - \omega)$. Suppose that the probability $\omega^* p_2 = \omega$ can be negligible and since $r = \gamma z$, $\mathcal{A}$ can compute the solution of DLP problem $z = r/\gamma = p_1/\gamma(\omega^* p_2 - \omega)$. However DLP hardness assumption cannot be solved in probabilistic polynomial time, there does not exist an adversary $\mathcal{A}$ who has a non-negligible advantage in the *Dishonest-User-2* game. □

5.3 Collusion Resistance

Theorem 5. *If the DLP difficulty problem holds, the proposed AR-CP-ABE scheme with user revocation is secure against user collusion in the selective model.*

Proof. Suppose that there exists a PPT adversary $\mathcal{A}$ who can break the proposed scheme with a non-negligible advantage ε after q_1 Type-I queries and q_2 Type-II queries, then we can construct a challenger $\mathcal{C}$ that can solve the DLP problem (g, g^z) with the advantage at most $\varepsilon/(q_1 \cdot q_2)$. Furthermore, in order to utilize $\mathcal{A}$ to obtain z, $\mathcal{C}$ should simulate a challenger for $\mathcal{A}$. Then, $\mathcal{C}$ interacts with $\mathcal{A}$ as follows:

- **Initialization:** $\mathcal{A}$ chooses a challenge access policy $W^* = (M^*, \rho^*, T^*)$ and a revocation list $\mathcal{R}^*$, where M^* is an $l^* \times n^*$ matrix and $n^* \leq q$, ρ^* is a mapping from rows of M^* to the attribute name, and $T^* = \{t_{\rho^*(i)}\}_{i \in [1, l^*]}$ is the attribute value related to (M^*, ρ^*).
- **Setup:** $\mathcal{C}$ generates the public parameters by calling the Setup algorithm and sends the public parameters PP to $\mathcal{A}$. Note that for each node of the binary tree $\mathcal{T}$, select $\{x_i\}_{i=0}^{2|\mathcal{U}|-2} \in \mathbb{Z}_p^*$ randomly. If a user $U \notin \mathcal{R}^*$, $\mathcal{C}$ sets $A = g^z$ and computes $\{y_i = A^{x_i} = g^{z x_i}\}_{i \in path(i_d)}$. Otherwise, $\mathcal{C}$ computes $\{y_i = g^{x_i}\}_{i \in path(i_d)}$.
- **Phase 1:** $\mathcal{C}$ first sets two empty lists L_I and L_{II}. $\mathcal{A}$ submits some queries as follows.
  - **Type-I key query** $\langle U_I, \mathcal{S}_I \rangle$: User U_I already has been revoked, but her or his attribute set $\mathcal{S}_I = (I_S, S) \in W^*$, where I_S and $S = \{s_i\}_{i \in I_S}$ are the attribute name and attribute value of the user. Firstly, $\mathcal{A}$ computes $H = h^\omega$ and

Table 1: Functionality comparisons

schemes	Revocation	Update	Collusion Resistance	Authority Accountability	Hidden Policy	Backward Security
Li *et al.* [11]	×	×	–	√	√	–
Vaanchig *et al.* [29]	attribute	ciphertext, key	√	×	×	√
ATIR-CPABE [16]	user	key	√	√	×	×
Han *et al.* [17]	user	ciphertext	×	×	√	×
Zhang *et al.* [18]	×	×	–	×	√	–
Ours	user	ciphertext	√	√	√	√

Table 2: Efficiency comparisons

schemes	KeyGen	Encrypt	Decrypt	Trace	Update
Li *et al.* [11]	$(7+3s)E+(1+2s)M$	$(4+5l)E$ $+(3+2l)M$	$(1+3l)P+5E$ $+(5+2l)M$	–	–
Vaanchig *et al.* [29]	$(1+s)E+M$	$(1+5l)E$ $+M$	$(2n+1)P+nE$ $+(2+n)M$	–	$2n_cE$
ATIR-CPABE [16]	$(10+s)E+(9+s)M$	$(6+3l)E$ $+(1+l)M$	$(5+2n)P+(6+n)E$ $+(6+n)M$	$(8+2s)P+7E$ $+4M$	$4E+3M$
Han *et al.* [17]	$(6+s)E+(1+s)M$	$(3+4l+r)E$ $+(1+l)M$	$(2+3n)P+(3+n)E$ $+(5+2n)M$	$(6+s)P+(2+s)E$ $+(3+s)M$	t_1E
Zhang *et al.* [18]	$(3+2s)E+(3+2s)$	$(4+6l)E$ $+(2+6l)M$	$(2n+1)P+nE+nM$	–	–
Ours	$(6+s+j)E+(1+s)M$	$(3+4l+r)E$ $+(1+l)M$	$(2+3n)P+(4+n)E$ $+(5+2n)M$	$(6+s)P+(3+s)E$ $+(3+s)M$	t_1E

An exponent operation in $\mathbb{G}_T, \mathbb{G}$ is represented by E. A bilinear pairing operation is represented by P. A multiplication is represented by M. The number of attributes that the access policy contains is represented by l. The number of attributes that the user owns is represented by s. The number of attributes that meets the access policy is represented by n. The number of $cover(R)$ is represented by r. The length of $path(U)$ is represented by j.

gives a zero-knowledge proof to $\mathcal{C}$ for requesting the intermediate keys. In Goyal's scheme [27], a simulator can use a knowledge extractor to extract ω. Thus $\mathcal{C}$ can use this technology to obtain ω. Then $\mathcal{C}$ can generate a intermediate key in the following:

1) Choose $c, r_I \in \mathbb{Z}_p$ randomly, and compute $K', K, L, L', L'', \{K_\tau\}_{\tau \in I_S}$ as follows:

$$\begin{aligned} K' &= c, & K &= g^{\frac{\alpha}{a+c}} h^{r_I \omega}, \\ L &= g^{r_I}, & L' &= (g^{r_I})^a = g^{a r_I}, \\ L'' &= g^{r_I x_{I,i_d}}, & K_\tau &= g^{x_{I,i_d} \cdot s_\tau \cdot r_I} \cdot u^{-(a+c) \cdot r_I}, \end{aligned}$$

where $\tau \in I_S$.

2) Suppose that $path(i_d) = \{i_0, \cdots, i_d\}$, where $i_0 = root$ and i_d is the value of the leaf node related to the user U_I. $\mathcal{C}$ computes $\{KU_{I,i} = g^{r_I \cdot \frac{x_{I,i_d}}{x_{I,i}}}\}_{i \in path(i_d)}$. Finally, $\mathcal{C}$ sends the key $(K', K, L, L', L'', \{KU_{I,i}\}_{i \in path(i_d)}, \{K_\tau\}_{\tau \in I_S}, \mathcal{S})$ to $\mathcal{A}$ and adds it into L_I.

– **Type-II key query** $\langle U_{II}, \mathcal{S}_{II} \rangle$: User U_{II} is unrevoked, but her or his attribute set $\mathcal{S}_{II} \notin W^*$. Then $\mathcal{C}$ chooses $c, r_{II} \in \mathbb{Z}_p$ randomly and generates a intermediate key by running the KeyGen algorithm as follows.

$$\begin{aligned} K' &= c, & K &= g^{\frac{\alpha}{a+c}} \cdot h^{r_{II} \cdot \omega}, & L &= g^{r_{II}}, \\ L' &= g^{a \cdot r_{II}}, & L'' &= (g^z)^{r_{II} \cdot x_{II,i_d}}, \\ K_\tau &= (g^z)^{x_{II,i_d} \cdot s_\tau \cdot r_{II}} \cdot u^{-(a+c) \cdot r_{II}}, \end{aligned}$$

$$\{KU_{II,i} = g^{r_{II} \cdot \frac{x_{II,i_d}}{x_{II,i}}}\}_{i \in path(i_d)}.$$

Finally, $(K', K, L, L', L'', \{KU_{II,i}\}_{i \in path(i_d)}, \{K_\tau\}_{\tau \in I_S}, \mathcal{S})$ will be sent to $\mathcal{A}$ and added to L_{II} by $\mathcal{C}$.

- **Challenge:** $\mathcal{A}$ sends two equal-length messages m_0, m_1 to $\mathcal{C}$. Then $\mathcal{C}$ flips a coin $\bar{b} \in \{0,1\}$ randomly and computes a ciphertext of $m_{\bar{b}}$ as follows:

1) Select $k_i, s \in \mathbb{Z}_p$ randomly, where $i \in [1, l]$, and calculate a partial ciphertext encrypted by the access policy W^*: $(C = m_{\bar{b}} \cdot e(g,g)^{\alpha s}, C_0 = g^s, C_0' = g^{as}, \{C_{i,1} = h^{\lambda_i} u^{k_i}, C_{i,2} = g^{-k_i \cdot t_{\rho(i)} + \lambda_i}, C_{i,3} = g^{k_i}\}_{i \in [1,l]})$.

2) Set the other ciphertext component ($\{T_j = y_j^s =$

$g^{zx_j s}\}_{j\in cover(\mathcal{R}^*)}$), which is related to the revocation list $\mathcal{R}^*$.

- **Phase 2:** Phase 2 is as same as Phase 1.
- **Guess:** If the challenge ciphertext can be decrypted by $\mathcal{A}$, he has to combine $K', K, L, L', L'', \{K_\tau\}_{\tau\in I_S}$ of U_I and $\{KU_{II,i}\}_{i\in path(i_d)}$ of U_{II}. Then $\mathcal{C}$ can successfully select matching tuples $K', K, L, L', L'', \{K_\tau\}_{\tau\in I_S}$ and $\{KU_{II,i}\}_{i\in path(i_d)}$ from L_I and L_{II}. Hence, he can compute

$$B = e(KU_{II,j}, T_j)^{T'} = e(g^{r_{II}\cdot\frac{x_{II,i_d}}{x_j}}, y_j^s)^\omega$$
$$= e(g^{r_{II}\cdot\frac{x_{II,i_d}}{x_j}}, g^{zx_j s})^\omega = e(g,g)^{r_I\cdot x_{I,i_d}\cdot s\cdot\omega}.$$

Therefore, only if $r_{II}\cdot x_{II,i_d}\cdot z = r_I\cdot x_{I,i_d}$, the above equation holds and the ciphertext can be decrypted correctly. Finally, $\mathcal{C}$ outputs $z = \frac{r_I\cdot x_{I,i_d}}{r_{II}\cdot x_{II,i_d}}$ as his answer.

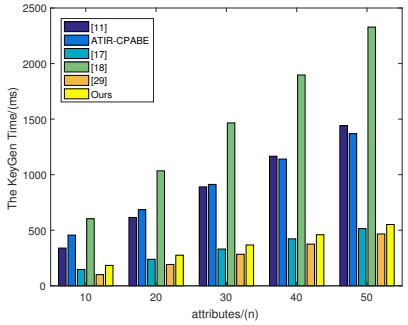
Suppose that $\mathcal{A}$ issues q_1 Type-I key queries and q_2 Type-II key queries, then the probability that $\mathcal{C}$ selects matching tuples is $1/(q_1\cdot q_2)$. Thus the advantage of $\mathcal{C}$ is at most $\varepsilon/(q_1\cdot q_2)$. □

6 Performance Analysis

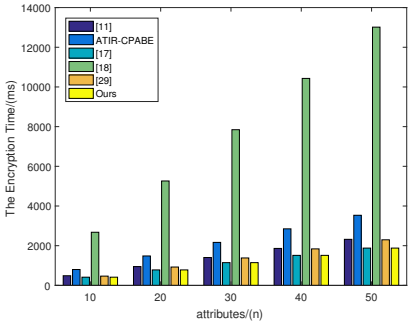
In this section, we will compare the functionality and evaluate the efficiency between the proposed scheme and the existing schemes [11, 16–18, 29].

6.1 Functionality Comparisons

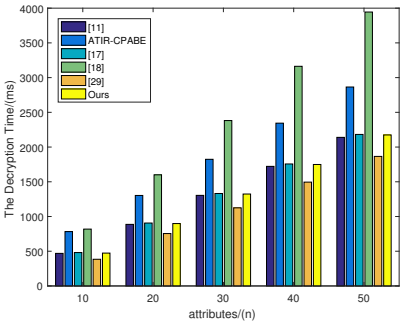
Table 1 shows that Ning *et al.*'s implicitly revocable CP-ABE scheme (ATIR-CPABE) [16] and the proposed AR-CP-ABE scheme can support the authority and user accountability and user revocation, while Han *et al.*'s scheme [17] and Zhang *et al.*'s [18] scheme can achieve hidden policy merely. Vaanchig *et al.* [29] can also implement user revocation and Li *et al.*'s scheme [11] can also realize the authority accountability. However, Vaanchig *et al.* [29] cannot trace the malicious user and Li *et al.*'s scheme [11] cannot remove the malicious users from the E-health system. Furthermore, Ning *et al.*'s scheme [16] cannot implement the backward security and the hidden policy, Han *et al.*'s scheme [17] cannot support the accountability and the backward security. At the same time, Ning *et al.*'s [16] and Han *et al.*'s [17] schemes could be vulnerable to user collusion attacks. Fortunately, the proposed AR-CP-ABE scheme can implement these functionalities at the same time and avoid the above security flaws.



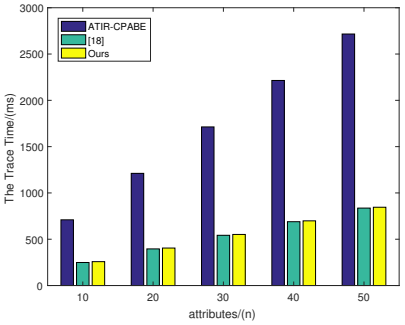
(a) KeyGen time



(b) Encryption time



(c) Decryption time



(d) Trace time

Figure 3: The comparisons of the results

6.2 Efficiency Analysis

In Table 2, we denote $t_1 = \sum_{j'\in cover(R')}(dep(j')-1-dep(j))$ and n_c as the number of ciphertexts including an attribute in its access structure. Table 2 shows that in the KeyGen and Trace algorithm, since the pairing operation takes more time than the exponent operation, the proposed scheme is more efficient than Li *et al.*'s scheme [11] and Ning *et al.*'s ATIR-CPABE scheme [16] in Trace al-

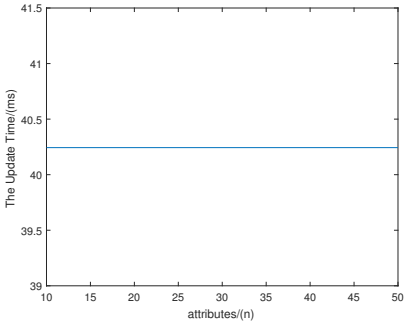


Figure 4: The efficiency of the update

gorithm. In the update algorithm, ATIR-CPABE scheme [16] and Vaanchig *et al.* [29] needs to generate the updated keys for all the unrevoked users, while only the ciphertext needs to be updated in the proposed scheme, thus the update time cannot be compared. It is pointed out that the proposed scheme can implement the accountability, user revocation and policy hiding at the same time, but the efficiency of the proposed scheme is comparable to that of Han *et al.*'s scheme [17].

Furthermore, Figure 3 demonstrates the efficiency test about the proposed scheme and related schemes and Figure 4 shows the efficiency of the Update algorithm with the number of attributes from 10 to 50. The machine for execution is 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz with 16.0GB RAM running 64 bits Windows 10. We set the attribute number from 10 to 50 and the revocation list from $\mathcal{R} = \varnothing$ to $\mathcal{R}^* = \{U_6\}$. Figure 3 vividly shows the comparisons of the KeyGen time, the Encryption time, Decryption time and the Trace time, respectively. It is clear that the proposed scheme is more efficient than other schemes [11, 16, 18]. Figure 4 shows that the update time is independent of the attributes in our scheme, which only updates the ciphertexts.

7 Conclusions

In this paper, we have presented a collusion resistance CP-ABE scheme with accountability, revocation and policy hiding. By binding together the secret value of the binary tree decryption node to the specific information of users, the proposed scheme can avoid user collusion attacks and achieve the backward security. At the same time, our scheme can implement the white-box accountability by embedding the secret value of user in the key and the partial hidden policy. Furthermore, the proposed scheme is proved to be secure under the decisional q-BDHE hardness assumption in the standard model. In the future, we aim to construct an accountable and revocable CP-ABE scheme with full hidden policy and use fine-grained attribute revocation to manage user permissions.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grants No.61807026, the National Key R&D Program of China under Grant No.2017YFB0802000, the Natural Science Basic Research Plan in Shaanxi Province of China under Grant No.2019JM-198, the Plan For Scientific Innovation Talent of Henan Province under Grant No.184100510012, and in part by the Program for Science and Technology Innovation Talents in the Universities of Henan Province under Grant No.18HASTIT022.

References

[1] A. Sahai and B. Waters, "Fuzzy identity-based encryption." The Advances in Cryptology-EUROCRYPT, Springer, 2005, pp. 457-473.

[2] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption." The 2007 IEEE Symposium on Security and Privacy, IEEE, 2007, pp. 321-334.

[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data." The 13th ACM Conference on Computer and Communications Security, ACM, 2006, pp. 89-98.

[4] Y. Zhang, R. H. Deng, S. Xu, J. Sun, Q. Li, and D. Zhang, "Attribute-based encryption for cloud computing access control: a survey." ACM Computing Surveys, Vol. 53, no. 4, pp. 83:1-83:41, 2020.

[5] Z. Liu, Z. Cao , and D. S. Wong, "Blackbox traceable CP-ABE: how to catch people leaking their keys by selling decryption devices on eBay." The 2013 ACM SIGSAC Conference on Computer and Communications, ACM, 2013, pp. 475-486.

[6] J. Li, K. Ren, B. Zhu, and Z. Wan, "Privacy-aware attribute-based encryption with user accountability." The 12th International Conference on Information Security, Springer, 2009, pp. 347-362.

[7] Z. Liu, Z. Cao, and D. S. Wong, "White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures." IEEE Transactions on Information Forensics and Security, Vol. 8, no. 1, pp.76-88, 2013.

[8] J. Ning, Z. Cao, X. Dong, L. Wei, and X. Lin, "White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes." IEEE Transactions on Information Forensics and Security, Vol. 10, no. 6, pp.1274-1288, 2015.

[9] J. Ning, X. Dong, and Z. Cao, "Accountable authority ciphertext-policy attribute-based encryption with white-box traceability and public auditing in the cloud." The 20th European Symposium on Research in Computer Security, Springer, 2015, pp. 270-289.

[10] Y. Zhang, J. Li, D. Zheng, X. Chen, and H. Li, "Towards privacy protection and malicious behavior

traceability in smart health." Personal and Ubiquitous Computing, Vol. 21, no. 5, pp. 851-830, 2017.
[11] J. Li, Y. Zhang, J. Ning, X. Huang, G. S. Poh, and D. Wang, "Attribute based encryption with privacy protection and accountability for cloudIoT." IEEE Transactions on Cloud Computing, doi: 10.1109/TCC.2020.2975184.
[12] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation." The 5th ACM Symposium on Information, Computer and Communications Security, ACM, 2010, pp. 261-270.
[13] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems." IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 7, pp. 1214-1221, 2011.
[14] J. Li, W. Yao, J. Han, Y. Zhang, and J. Shen, "User collusion avoidance CP-ABE with efficient attribute revocation for cloud storage." IEEE Systems Journal, Vol. 12, no. 2, pp. 1767-1777, 2018.
[15] K. Lee, S. G. Choi, D. H. Lee,J. H. Park, and M. Yung, "Self-updatable encryption: time constrained access control with hidden attributes and better efficiency." The Advances in Cryptology-ASIACRYPT, Springer, 2013, pp. 235-254.
[16] J. Ning, Z. Cao, X. Dong, K. Liang, L. Wei, and K. K. R. Choo, "CryptCloud+: secure and expressive data access control for cloud storage." IEEE Transactions on Services Computing, Vol. 14, no. 1, pp. 111-124, 2021.
[17] D. Han, N. Pan, and K. C. Li, "A traceable and revocable ciphertext-policy attribute-based encryption scheme based on privacy protection." IEEE Transactions on Dependable and Secure Computing, doi: 10.1109/TDSC.2020.2977646.
[18] Y. Zhang, D. Zheng, and R. H. Deng, "Security and privacy in smart health: efficient policy-hiding attribute-based access control." IEEE Internet of Things Journal, Vol. 5, no. 3, pp. 2130-2145, 2018.
[19] L. Sun, and C. Xu, "Hidden policy ciphertext-policy attribute based encryption with conjunctive keyword search." The 3rd IEEE International Conference on Computer and Communications, IEEE, 2017, pp. 1439-1443.
[20] D. Naor, M. Naor, and J. Lotspiech, "Revocation and tracing schemes for stateless receivers." The Advances in Cryptology-CRYPTO, Springer, 2001, pp. 41-62.
[21] B. Waters, "Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization." The Public Key Cryptography-PKC, Springer, 2011, pp. 53-70.
[22] D. Boneh andk B. Xavier, "Short signatures without random oracles." The Advances in Cryptology-EUROCRYPT, Springer, 2004, pp. 56-73.
[23] B. Smith, "Isogenies and the discrete logarithm problem in jacobians of genus 3 hyperelliptic curves." The Advances in Cryptology-EUROCRYPT, Springer, 2008, pp. 163-180.
[24] M. Bahrami, and M. Singhal, "A dynamic cloud computing platform for eHealth systems." The 17th International Conference on E-health Networking, Application and Services (HealthCom), IEEE, 2015, pp. 435-438
[25] Z. Liu, F. Yin, J. Ji, and B. Wang, "Revocable and searchable attribute-based encryption scheme with multi-keyword and verifiability for internet of things." International Journal of Network Security, Vol. 23, no. 2, pp. 205-219, 2021.
[26] Z. Liu, Y. Liu, J. Xu, and B. Wang, "Verifiable attribute-based keyword search encryption with attribute revocation for electronic health record system." International Journal of Network Security, Vol. 22, no. 5, pp. 845-856, 2020.
[27] V. Goyal, "Reducing trust in the PKG in identity based cryptosystem." The Advances in Cryptology-CRYPTO, Springer, 2007, pp. 430-447.
[28] M. S. Hwang, T. H. Sun, and C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service." Journal of Circuits Systems and Computers, Vol 26, No. 5, 1750072: 1-16, 2016.
[29] N Vaanchig, H. Xiong, W. Chen, and Z. Qin, "Achieving collaborative cloud data storage by key-escrow-free multi-authority CP-ABE scheme with dual-revocation." International Journal of Network Security, Vol 20, No.1, PP. 95-109, 2018.

Biography

Zhenhua Liu received the B.S. degree from Henan Normal University in 2000, and the M.S. and Ph.D. degrees from Xidian University, China, in 2003 and 2009, respectively. He is currently a Professor of Xidian University, China. His current research interests include cryptography and information security.

Yingying Ding received the B.S. degree from Henan Normal University in 2019. She is currently going in for the M.S. degree in mathematics with Xidian University, China. Her research interests concentrate on cryptography and cloud security.

Ming Yuan received the B.S. degree from Henan Normal University in 2018. She is currently going in for the M.S. degree in mathematics with Xidian University, China. Her research focuses on network and information security .

Baocang Wang received the B.S., the M.S. and Ph.D. degrees from Xidian University, China, in 2001, 2004, and 2006, respectively. He is currently a Professor with the State Key Laboratory of Integrated Services Networks of Xidian University, China. His research focuses on post-quantum cryptography, number theoretic algorithms, and cloud security.

Analysis of Two Outsourcing Algorithms for Solving Quadratic Congruence

Lihua Liu and Yujie Li
(Corresponding author: Lihua Liu)
Department of Mathematics, Shanghai Maritime University
Haigang Ave 1550, Shanghai, 201306, China
Email: liulh@shmtu.edu.cn
(Received Oct. 16, 2021; Revised and Accepted Mar. 23, 2022; First Online Apr. 11, 2022)

Abstract

We show that the outsourcing algorithms [IEEE ITJ, 7(4), 2020, 2968–2981] for solving quadratic congruence in the Internet of Things are flawed. (1) The Cipolla algorithm is unsuitable for the discussed scenario. The underlying modulus is generally a composite containing two strong primes to resist some factorization algorithms. Besides, the Rabin cryptosystem explicitly requires that $p \equiv q \equiv 3 \bmod 4$. In this case, the Cipolla algorithm is unnecessary. (2) The outsourcer can finish the computation solely, even if $p \not\equiv 3 \bmod 4$ and $q \not\equiv 3 \bmod 4$. He doesn't have to outsource the original problem because he must pay out equal-cost $O(\log^3 p)$ in the proposed outsourcing scenario.

Keywords: Adleman-Manders-Miller algorithm; Cipolla Algorithm; Pollard Method; Quadratic Congruence; Strong Prime

1 Introduction

The Internet of Things (IoT) consists of a large amount of resource constrained devices to collect and compute data. These devices need to execute some public-key cryptographic protocols for confidentiality and authentication [2]. But some public-key computations are too expensive for these devices to finish. It becomes usual for these resource constrained devices to outsource those heavy computations to cloud or edge servers.

In the outsourcing scenario, one has to tackle some security challenges [14]. Usually, it requires that:

- The sensitive information contained in the outsourced data should not be exposed to the cloud servers.
- The outsourcer can verify the correctness of the returned results.
- The outsourcer can save much computational cost in comparison with the incurred communication cost.

Dreier and Kerschbaum [10] have put forth a method for secure outsourcing of linear programming. After that, Wang *et al.* [21] proposed a scheme for outsourcing large-scale systems of linear equations. But its homomorphic encryption system [15] was not compatible with Jacobi iteration [3]. In 2014, Chen *et al.* [7] presented one algorithm for outsourcing linear regression problem, but neglected to check whether the client can solve the original problem solely [4].

In 2015, Salinas *et al.* [18, 19] have presented an outsourcing scheme for large-scale sparse linear systems of equations. In 2018, Ding *et al.* [9] pointed out that in the Salinas *et al.*'s scheme the cloud server can recover a client's input. In 2020, Cao and Markowitch [5] argued that in the discussed scenario it was unnecessary for a client to outsource the problem because he can finish the computations solely. Recently, Wang *et al.* [11, 22] have presented a survey for reversible data hiding for VQ-compressed images. Pan *et al.* [8, 12, 13, 16, 17, 20] put forth some batch verification schemes for identifying illegal signatures, smart card-based password authentication schemes, and data collaboration scheme with hierarchical attribute-based encryption in cloud computing scenario.

The Rabin cryptosystem is based on the intractability of solving $x^2 \equiv \bmod n$, where n is an RSA modulus. It has been proved that the security of Rabin cryptosystem was equivalent to factoring n, while the security of RSA has not yet been proven. So, in some cases the Rabin cryptosystem is more appreciated because the encryptor only needs to do one multiplication modulo n. For example, the IoT security framework makes use of the Rabin encryption to offer confidentiality.

Table 1: Cipolla algorithm

Let p be an odd prime, $n \in \mathbb{F}_p$ be a quadratic residue.
Find $a \in \mathbb{F}_p$ such that $(a^2 - n)^{\frac{p-1}{2}} \equiv -1 \bmod p$. Compute the quadratic root $x = (a + \sqrt{w})^{\frac{p+1}{2}} \bmod p$ within the filed $F_p(\sqrt{w})$, where $w = a^2 - n$.

The Cipolla algorithm (Table 1) can be used to solve quadratic congruences. Recently, Zhang *et al.* [23] have presented two outsourcing algorithms based on Cipolla algorithm. In this note, we show that the outsourcing algorithms have two flaws.

2 Review of the Algorithms

Given the odd prime p and $n \in \mathbb{F}_p$, the client transforms the two numbers into $p' = pq, n' = n - r_1 p$, where q, r_1 are two random blinders. We now only describe the second outsourcing algorithm as below (Table 2). Its correctness is based on that

$$\begin{aligned} x \equiv & (a + \sqrt{a^2 - n'})^k R'_2 \bmod p \\ \equiv & (a + \sqrt{a^2 - n'})^k ((a + \sqrt{a^2 - n'})^{d'_2} \bmod p') \bmod p \\ \equiv & (a + \sqrt{a^2 - n'})^{k + d'_2} \bmod p \\ \equiv & (a + \sqrt{a^2 - n'})^{(p+1)/2} \bmod p \\ & ((a^2 - n')^{d'} \bmod p') \bmod p \\ & \equiv (a^2 - n')^{(p-1)/2 + r_2(p-1)} \bmod p \\ & \equiv (a^2 - n')^{(p-1)/2} \bmod p \\ & \equiv (a^2 - n)^{(p-1)/2} \bmod p \equiv -1. \end{aligned}$$

3 Analysis

In general, an outsourcing algorithm should meet two basic requirements: privacy—nobody can know the client's input and output except himself; efficiency—the client can save much cost in comparison with solving the original problem solely. But we find the proposed outsourcing algorithms fail to meet the second requirement.

3.1 Strong Primes Should Be Chosen

In order to resist some factorization algorithms such as the Pollard $\rho + 1$ or $\rho - 1$ methods, PKCS suggests the using of strong primes. A strong prime p satisfies that $p - 1$ contains a large prime factor, and $p + 1$ also contains a large prime factor. The Rabin cryptosystem in particular requires that $p \equiv q \equiv 3 \bmod 4$. In this case, the user can simply recover the square root by computing $n^{\frac{p+1}{4}} \bmod p$. The claim that [page 2979, Ref. [23]] "in the Rabin cryptosystem, p and q are not necessary to be $p \equiv q \equiv 3 \bmod 4$" is incorrect. To the best of our knowledge, the Cipolla algorithm is unnecessary for a public key cryptographic scheme based on the intractability of factorization.

3.2 The Outsourcer Can Finish The Computation Solely

The outsourcing algorithms fail to save much cost for the outsourcer, even if $p \not\equiv 3 \bmod 4$. As we see, the outsourced task is just to do the computation of finding a such that

$$(a^2 - n)^{\frac{p-1}{2}} \equiv -1 \bmod p$$

For a randomly chosen $a \in \mathbb{F}_p$, the checking requires $O(\log^2 p)$ cost by computing the Legendre Symbol $\left(\frac{a^2-n}{p}\right)$. Assuming Extended Riemann Hypothesis, it requires $O(\log p)$ tests [1] to find out a quadratic nonresidue ρ such that $\rho^{\frac{p-1}{2}} \equiv -1 \bmod p$. The modular exponentiation

$$(a + \sqrt{w})^{\frac{p+1}{2}} \bmod p.$$

requires $O(\log^3 p)$ cost. Thus, the client only needs to pay $O(\log^3 p)$ cost to find the square root, if he tries to solve the problem solely.

In the outsourcing scenario the client needs to pay $O(\log k \log^2 p)$ cost to compute

$$(a + \sqrt{a^2 - n'})^k R'_2 \bmod p.$$

The cloud (not knowing the modulus p) should ultimately find out $a \in \mathbb{F}_{p'}$ such that

$$((a^2 - n')^{d'} \bmod p') \bmod p \equiv -1.$$

The procedure needs to do $O(\log p)$ interactions between the client and the cloud, which requires much cost because the communicators and transferred data should be authenticated. Usually, it requires at least $O(\log^2 p)$ cost (equal to doing a multiplication of two integers with the same length $\log p$) for the client to authenticate the transferred data in each loop. So, the client needs to pay out $O(\log^3 p)$ cost at least. See Table 3 for the cost comparisons. Thus, it is unnecessary for the client to outsource the original problem because he needs to pay out equal cost $O(\log^3 p)$ in the outsourcing scenario.

Other flaws. The communication cost analysis (Table II, [23]) neglects the cost for underlying communicators authentication and data integrity authentication. The listed references [16, 21] are misleading due to the shortcomings shown in [3, 5]. By the way, the expressions $R'_1 = (a^2 - n)^{d'}$ and $R'_1 = a^2 - n'$ (page 2975, Ref. [23]) are not computed over the ring $\mathbb{Z}_n$, which should be revised as $R'_1 = (a^2 - n)^{d'} \bmod p'$, where p' is a secret prime factor owned only by the outsourcer.

3.3 Further Discussions

There is a more efficient algorithm for root extraction, i.e., the Adleman-Manders-Miller algorithm [1] (Table 4). Its basic idea can be described as below. Write $p - 1 = 2^t s, 2 \nmid s$. Given a quadratic residue δ and a quadratic nonresidue ρ, i.e.,

$$(\delta^s)^{2^{t-1}} \equiv 1 \bmod p, \quad (\rho^s)^{2^{t-1}} \equiv -1 \bmod p.$$

If $t \geq 2$, then $(\delta^s)^{2^{t-2}} \bmod p \in \{1, -1\}$. Take $k_1 \in \{0, 1\}$ such that

$$(\delta^s)^{2^{t-2}} (\rho^s)^{2^{t-1} \cdot k_1} \equiv 1 \bmod p.$$

Table 2: SoSQC2

Client: $\{n, p\}$	Cloud
[Transformations] Pick $r_1, r_2 \in \mathbb{F}_p$, a short random integer k, and a large prime q. Compute $p' = pq, n' = n - r_1 p$, $d' = (p-1)/2 + r_2(p-1)$, $d'_2 = (p+1)/2 - k$. $\xrightarrow{n', d', d'_2, p'}$	
Check $R'_1 \equiv -1 \bmod p$. If it fails, ask for a new number until such an integer is found. $\xrightarrow{\text{OK}}$	[Quadratic nonresidue finding] Pick $a \in \mathbb{F}_p$, compute $R'_1 = (a^2 - n')^{d'} \bmod p'$. $\xleftarrow{R'_1}$ Upon receiving "OK", compute $R'_2 = (a + \sqrt{a^2 - n'})^{d'_2} \bmod p'$. $\xleftarrow{a, R'_2}$
[Retrieval] Compute $x = (a + \sqrt{a^2 - n'})^k R'_2 \bmod p$. Check that $x^2 \equiv n \bmod p$.	

Table 3: Cost comparisons

Unoutsourcing case	Outsourcing case
(1) Find $a \in \mathbb{F}_p$ such that $(a^2 - n)^{\frac{p-1}{2}} \equiv -1 \bmod p$, which requires $O(\log^3 p)$ cost.	(1) Do $O(\log p)$ interactions with the cloud to find $R'_1 \equiv -1 \bmod p$, which requires $O(\log^3 p)$ cost at least.
(2) Compute $(a + \sqrt{w})^{\frac{p+1}{2}} \bmod p$ which requires $O(\log^3 p)$ cost.	(2) Compute $(a + \sqrt{a^2 - n'})^k R'_2 \bmod p$ which requires $O(\log k \log^2 p)$ cost.

Since $(\delta^s)^{2^{t-3}} (\rho^s)^{2^{t-2} \cdot k_1} \bmod p \in \{1, -1\}$, take $k_2 \in \{0, 1\}$ such that

$$(\delta^s)^{2^{t-3}} (\rho^s)^{2^{t-2} \cdot k_1} (\rho^s)^{2^{t-1} \cdot k_2} \equiv 1 \bmod p.$$

Likewise, take $k_3, \cdots, k_{t-1} \in \{0, 1\}$ such that

$$\delta^s (\rho^s)^{2 \cdot k_1 + 2^2 \cdot k_2 + \cdots + 2^{t-1} \cdot k_{t-1}} \equiv 1 \bmod p.$$

Thus,

$$\left(\delta^{\frac{s+1}{2}}\right)^2 \left((\rho^s)^{k_1 + 2 \cdot k_2 + \cdots + 2^{t-2} \cdot k_{t-1}}\right)^2 \equiv \delta \bmod p.$$

Its computational complexity is $O(\log^3 p + t^2 \log^2 p)$ (see [6]). Since p is usually set as a strong prime, i.e., $t = 1$, it becomes $O(\log^3 p)$, and $\delta^{\frac{p+1}{2}} \equiv \delta \bmod p$.

If $4 \mid p + 1$, then $\delta^{\frac{p+1}{4}} \bmod p$ is just a square root of δ modulo p. This is the reason that the Rabin Cryptosystem specifies $p \equiv q \equiv 3 \bmod 4$. In this case, it avoids the need to find a quadratic nonresidue.

The complexities of Cipolla algorithm and Adleman-Manders-Miller algorithm are both dominated by the procedure to find a quadratic nonresidue. The Cipolla algorithm needs to find a special quadratic nonresidue which should be written as $a^2 - n$, while the Adleman-Manders-Miller algorithm only needs to find a common quadratic nonresidue ρ. So, the Adleman-Manders-Miller algorithm is more efficient than the Cipolla algorithm. Besides, the Adleman-Manders-Miller algorithm can be extended to r^{th} root extraction ($r > 2$).

Table 4: Adleman-Manders-Miller algorithm

Let p be an odd prime, $\delta \in \mathbb{F}_p$ be a quadratic residue.
Write $p - 1$ as $2^t s$, where $2 \nmid s$.
Find a quadratic nonresidue ρ, i.e., $\rho^{\frac{p-1}{2}} \equiv -1 \bmod p$.
Take $k_1, \cdots, k_{t-1} \in \{0, 1\}$, such that $\delta^s (\rho^s)^{2 \cdot k_1 + 2^2 \cdot k_2 + \cdots + 2^{t-1} \cdot k_{t-1}} \equiv 1 \bmod p$.
Compute the quadratic root $x \equiv \delta^{\frac{s+1}{2}} (\rho^s)^{k_1 + 2 \cdot k_2 + \cdots + 2^{t-2} \cdot k_{t-1}} \bmod p$.

4 Conclusion

We show that the Zhang *et al.*'s outsourcing algorithms cannot save much cost for the client. We want to stress that the Adleman-Manders-Miller algorithm is more efficient than the Cipolla algorithm because the latter needs to find out a special quadratic nonresidue.

Acknowledgements

We thank the National Natural Science Foundation of China (Project 61411146001). We are grateful to the reviewers for their valuable suggestions.

References

[1] L. Adleman, K. Manders, and G. Miller, "On taking roots in finite fields," in *Proceedings of 18th Annual Symposium on Foundations of Computer Science*, pp. 175–178. IEEE Computer Society, 1977.

[2] E. Bresson, D. Catalano, and D. Pointcheval, "A simple public-key cryptosystem with a double trapdoor decryption mechanism and its applications," in *Proceedings of 9th International Conference on the Theory and Application of Cryptology and Information Security, ASIACRYPT 2003*, pp. 37–54, Taipei, December 2003.

[3] Z. J. Cao and L. H. Liu, "Comment on 'harnessing the cloud for securely outsourcing large-scale systems of linear equations'," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 5, pp. 1551–1552, 2016.

[4] Z. J. Cao, L. H. Liu, and O. Markowitch, "Comment on 'highly efficient linear regression outsourcing to a cloud'," *IEEE Transactions on Cloud Computing*, vol. 7, no. 3, p. 893, 2019.

[5] Z. J. Cao and O. Markowitch, "Comment on 'efficient secure outsourcing of large-scale sparse linear systems of equations'," *IEEE Transactions on Big Data*, 10.1109/TBDATA.2020.2995200.

[6] Z. J. Cao, Q. Sha, and X. Fan, "Adleman-Manders-Miller root extraction method revisited," in *Proceedings of 7th International Conference on Information Security and Cryptology*, pp. 77–85. Springer, 2011.

[7] F. Chen and *et al.*, "Highly efficient linear regression outsourcing to a cloud," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 499–508, 2014.

[8] Y. H. Chen and *et al.*, "Research on the secure financial surveillance blockchain systems," *International Journal of Network Security*, vol. 22, no. 4, pp. 708–716, 2020.

[9] Q. Ding and *et al.*, "Efficient and secure outsourcing of large-scale linear system of equations," *IEEE Transactions on Cloud Computing*, 10.1109/TCC.2018.2880181.

[10] J. Dreier and F. Kerschbaum, "Practical privacy-preserving multiparty linear programming based on problem transformation," in *Proceedings of IEEE International Conference on Privacy, Security, Risk, and Trust, and IEEE International Conference on Social Computing*, pp. 916–924, Boston, USA, Oct. 2011.

[11] L. C. Huang, T. Y. Chang, and M. S. Hwang, "A conference key scheme based on the Diffie-Hellman key exchange," *International Journal of Network Security*, vol. 20, no. 6, pp. 1221–1226, 2018.

[12] L. H. Liu and *et al.*, "A note on one secure data self-destructing scheme in cloud computing," *International Journal of Network Security*, vol. 22, no. 1, pp. 36–40, 2020.

[13] L. H. Liu and L. M. Hong, "Analysis of one authenticated key agreement scheme for consumer usb mass storage devices resilient to unauthorized file decryption," *International Journal of Electronics and Information Engineering*, vol. 13, no. 1, pp. 10–16, 2021.

[14] D. Marinescu, *Cloud Computing Theory and Practice*. USA: Elsevier, 2013.

[15] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proceeding of International Conference on the Theory and Application of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT 1999*, pp. 223–238, Prague, Czech Republic, May 1999.

[16] H. Pan and *et al.*, "Research on batch verification schemes for identifying illegal signatures," *International Journal of Network Security*, vol. 21, no. 6, pp. 1062–1070, 2019.

[17] H. T. Pan, H. W. Yang, and M. S. Hwang, "An enhanced secure smart card-based password authentication scheme," *International Journal of Network Security*, vol. 22, no. 2, pp. 358–363, 2020.

[18] S. Salinas and *et al.*, "Efficient secure outsourcing of large-scale sparse linear systems of equations," *IEEE Trans. Big Data*, vol. 4, no. 1, pp. 26–39, 2018.

[19] S. Salinas, C. Luo, X. Chen, and P. Li, "Efficient secure outsourcing of large-scale linear systems of equations," in *Proceedings of 2015 IEEE Conference on Computer Communications, INFOCOM 2015*, pp. 1035–1043, Hong Kong, Apr. 2015.

[20] W. L. Tai, Y. F. Chang, and W. H. Huang, "Security analyses of a data collaboration scheme with hierarchical attribute-based encryption in cloud computing," *International Journal of Network Security*, vol. 22, no. 2, pp. 212–217, 2020.

[21] C. Wang and *et al.*, "Harnessing the cloud for securely outsourcing large-scale systems of linear equations," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1172–1181, 2013.

[22] Y. L. Wang, J. J. Shen, and M. S. Hwang, "A survey of reversible data hiding for VQ-compressed images," *International Journal of Network Security*, vol. 20, no. 1, pp. 1–8, 2018.

[23] H. Zhang and *et al.*, "Practical and secure outsourcing algorithms for solving quadratic congruences in internet of things," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2968–2981, 2020.

Biography

Lihua Liu, associate professor with Department of Mathematics at Shanghai Maritime University, received her PhD degree in applied mathematics from Shanghai Jiao

Tong University. Her research interests include combinatorics and cryptography.

Yujie Li is currently pursuing her master degree from Department of Mathematics at Shanghai Maritime University. Her research interests include information theory and applied mathematics.

Analysis of Policy Anomalies in Distributed Firewalls

Yu-Zhu Cheng[1] and Qiu-ying Shi[2]
(Corresponding author: Yu-Zhu Cheng)
School of Software, Changsha Social Work College[1]
Changsha 410004, China
(Email: vogue21ct@qq.com)
School of Computer Science and Engineering, Central South University[2]
Changsha 410083, China
(Received Oct. 17, 2021; Revised and Accepted Mar. 12, 2022; First Online Apr. 11, 2022)

Abstract

In order to solve the problem of anomaly analysis in distributed firewalls, a rule anomaly detection method based on spatial relationship comparison is proposed. Firstly, all firewall rules on the network path are mapped into the Firewall Design Matrix(FDM) in reverse order to form independent unit space sets. Secondly, the unit space overlap of all the upstream firewalls corresponding to the most downstream firewall FW_d is obtained. Then this overlap is mapped to the rule space of FW_d, where the uncovered area of FW_d is the desired shadowing anomaly. For spuriousness anomaly, we first calculate the unit space overlap of all downstream firewalls corresponding to the most upstream firewall FW_u and then map the overlap to the rule space of FW_u, where the uncovered area of FW_u is the desired spuriousness anomaly. Simulation results show that this method can accurately and efficiently detect all the rule shadowing anomalies and spuriousness anomalies.

Keywords: Anomaly Detecting; Distributed Firewall; Firewall Policy; Policy Anomaly Analysis

1 Introduction

Firewalls are critical components of network security and are deployed at the entrances between a private network and the Internet to monitor all incoming and outgoing packets. The function of a firewall is to examine the field values of every packet and decide whether to accept or discard a packet according to the firewall policies. The policy is specified as a sequence of rules, each of which has a predicate over some packet header fields and a decision to be performed upon the packets that match the predicate. With the rapid development of the Internet, it is more and more difficult to efficiently manage firewall rules as the number of rules increases. It is known that the rules in a firewall policy are logically entangled because of conflicts among rules and the resulting order sensitivity [23]. Ordering the rules correctly in a firewall is critical and difficult.

In a traditional perimeter firewall environment, the local firewall policy may include intra-firewall anomalies, where the same packet may match multiple filtering rules. Moreover, in distributed firewall environment, firewalls might also have inter-firewall anomalies when individual firewalls in the same path perform different filtering actions on the same traffic [1]. Therefore, the administrator must give special attention not only to all rule relations in the same firewall in order to determine the correct rule order, but also to all relations between rules in different firewalls, in order to determine the proper rule placement in the proper firewall. In addition, a typical large-scale enterprise network might involve hundreds of rules that might be written by different administrators at different times. This significantly increases the potential of anomaly occurrence in the firewall policy, jeopardizing the security of the protected network. Therefore, the effectiveness of firewall security depends on the provision of policy analysis techniques that network administrators can use to analyze the correctness of written firewall filtering rules.

In this paper, we address the problem of anomaly analysis in distributed firewalls. Our work presents a significant contribution in this field since it offers a new approach to analyze anomalies within distributed firewall filtering rules, which is based on the complete definition of anomalies stated in the work of Al-Shaer and Hamed [1]. Our paradigm is based on a two-stage analysis process. In the first stage, for intra-firewall anomalies, we design an approach to eliminate them while maintaining the consistency, compactness and completeness of the original firewall rules [10]. In the second stage, we propose a new approach to analyze inter-firewall anomalies based on the comparison of rule spatial relation, this method can ac-

curately and efficiently discover the shadowing anomalies and spuriousness anomalies of distributed firewall policy. The approach is also very effective, performance analysis and simulation results show that the algorithm has a high execution efficiency.

The rest of this paper is organized as follows: the related work is presented in Section 2; then we define the intra-firewall anomalies, and present the algorithm for elimination of them in Section 3. In Section 4, we first give the classification and detection algorithm of inter-firewall anomalies, followed by the analysis of experimental results in Section 5. Finally, the conclusion is drawn in Section 6.

2 Related Work

Although distributed firewall policy analysis has been given strong attention in the research community, some excellent algorithms and corresponding tools are mostly focused on the problems of rule design, rule compression and rule conflict detection for traditional perimeter firewalls [8–10, 14, 16], while methods and tools for anomaly detection of distributed firewall policies are not many yet [17, 18, 22].

Nowadays, the design, optimization and management of firewall policies have attracted wide attention of researchers, the problem of firewall policy design is to design corresponding firewall rules according to the network security requirements described by natural language. At present, the common method is to define and analyze firewall security policies using specific design models, such as Trie binary tree [3], FDD [10], FDM [8] *et al*, and then generate filtering rules through policy mapping. In our prior work [8], an approach to designing firewall based on multidimensional matrix was proposed. Specifically, we developed a new designing model, namely firewall design matrix (FDM), and the corresponding construction algorithm for mapping firewall rules to FDM, whose consistency and compactness can be achieved by the construction algorithm, and then a firewall generation algorithm was proposed to generate the target firewall rules equivalent to the original ones while maintaining the completeness.

During the process of firewall filtering packets, when a packet matches two or more rules at the same time and the decision of these rules differs, rules conflict. At this time, data packets are processed according to the decision defined by the high priority rule. Generally speaking, rule conflicts need to be detected and eliminated as many as possible, otherwise some packets will be filtered incorrectly, which will bring some adverse consequences to the network. At present, most of the research on rule conflict focuses on the traditional perimeter firewall, including conflict classification, conflict detection and conflict elimination.

The classification of rule conflicts have a detailed discussion in [11]. The traditional perimeter firewall rule conflicts are classified as shadowing anomaly, correlation anomaly, generalization anomaly and redundancy anomaly. Rule conflict detection is to find all conflict rule pairs in a rule set, or to find all rules that conflict with a rule set. Conflict elimination technology refers to the rule set does not have any conflict rules after rule conflicts are eliminated. Literature [12] compares and analyzes the inconsistent parts of rule decision-making in each segment to detect rule conflicts and to eliminate them by adjusting the order of rules. But because rules often span multiple segments, it is possible to introduce anomalies to other segments when adjusting the order of rules in one segment. Moreover, in some cases, no matter how to adjust, it can not achieve the purpose of eliminating conflicts [2]. In addition, with the development of network applications, the number of firewall rules is increasing, it is too complicated to detect and eliminate conflicts manually by administrators, even though recent studies have been trying to resolve errors among polices and optimize a performance of firewall [4, 7, 21]. For this reason, it is important to reveal policy conflicts and potential problems automatically, and provide an intuitive solution to the network administrator. F.Chen *et al.* presented a method to automatically detect rule anomalies [6], they first defined a fault model includes five types of faults: wrong order, missing rules, wrong decisions, wrong predicates, and wrong extra rules. For each type of fault, they proposed a correction technique based on the passed and failed tests of a firewall policy,where the passed and failed tests are automated generated and classified based on packet generation and classify techniques [13]. The approach is effective to correct a faulty firewall policy with faults of wrong order, wrong decisions, and wrong extra rules, but it is not ideal for the correction results of two types of faults: missing rules and wrong predicates.

In distributed firewall environment, the local firewall policy may include intra-firewall anomalies, and might also have inter-firewall anomalies when individual firewalls in the same path perform different filtering actions on the same traffic. In [1], Al-Shaer and Hamed classified various intra-firewall and inter-firewall anomalies, prominent of which being Shadowing, Redundancy, Correlation and Generalization. Based on the specified anomalies, a Firewall Policy Advisor tool (FPA) was developed to detect the existing firewall anomalies in the specified network. The disadvantages of the tool include detection of only pair wise firewall anomalies. Similar to FPA, Lihua Yuan *et al.* [24] introduced Fireman, a toolkit for anomaly analysis in distributed firewalls, while it evaluates firewall configurations as a whole piece rather than just limiting to relation between two firewall rules. Chi-Shih Chao proposed an anomaly diagnosis system in which a RAR (Rule Anomaly Relation) tree is created based on ACL's [5]. The system detect inter- as well as intra-firewall anomalies in a feasible time range. Also, the system suggests network administrators regarding correction in behavior mismatching errors. In [19], Pedditi *et al.* presented a design of a new protocol namely FIEP (Firewall Information Exchange Protocol) which provides a communi-

cation mechanism for distributed firewalls to communicate with each other. Like the Border Gateway Protocol (BGP) that enables routers to exchange routing information, the firewalls can automatically check for inconsistencies in their firewall configuration through message passing. The disadvantages of the protocol include the need to change the existing enterprise hardware which is time consuming and expensive, difficult to implement in existing networks. In order to solve the problem of insufficient usability caused by the limitations of text interface and the complexity of practical use, K. Taeyong *et al.* presented a three-dimensional hierarchical visualization method F/Wvis for intuitive ACL management and analysis [15]. F/Wvis can provide in-depth user interface through hierarchical visualization method, support ACL management of large-scale networks and analysis of policy details and exceptions.

In this paper, we will discuss the inter- and intra-firewall anomalies in detail along with the definition of distributed firewall network model in [1], which is the basis of our anomaly analysis of the distributed firewall policies.

3 Analysis and Detection of Intra-firewall Anomalies

3.1 Intra-firewall Anomaly Definition

Packet classification is performed by sequentially matching the packet against firewall rules until a match is found. If the rules are independent of each other, the order between them is inessential. However, it is very common to have firewall rules interrelated while their decisions are different. In this case, the rules in a firewall policy are logically entangled because of conflicts and the resulting order sensitivity.

Therefore, an intra-firewall policy anomaly is defined as the existence of two or more filtering rules that may match the same packet, or the existence of a rule that can never match any packet that cross the firewall [10].

3.2 Intra-firewall Anomaly Classification

According to the definition of policy anomaly, the rule configuration anomaly can be classified as conflict, incomplete and redundancy. Take the firewall which has four rules as an example:

r_1:F_1∈[0,8] ∧ F_2∈[3,7] →*accept*,
r_2:F_1∈[0,9] ∧ F_2∈[3,9] →*discard*,
r_3:F_1∈[2,7] ∧ F_2∈[3,6] →*accept*,
r_4:F_1∈[0,8] ∧ F_2∈[0,3] →*accept*.

Although the number of rules in this firewall is small, it exemplifies all the three problems of firewall, namely consistency, completeness and compactness. Consistency means that the rules are ordered correctly, completeness means that every packet satisfies at least one rule in the firewall, and compactness means that the firewall has no redundant rules [10].

The two rules r_1 and r_2 are conflicting because there are packets whose fields satisfy the predicates of both r_1 and r_2 (for example, a packet with F_1∈[0,8] ∧ F_2∈[3,7] can satisfy the predicates of both r_1 and r_2) and these two rules have different decisions. Therefore, the relative order of these two rules with respect to one another in the sequence of rules becomes very critical. The relative order of rules r_1 and r_2 is likely a consistency error. The second error in the above rule sequence is that any packet with F_1∈[9,9] ∧ F_2∈[0,2] does not satisfy the predicate of any of the four rules. Such an error is referred to as a completeness error, which can be corrected by adding one new rule r_5:F_1∈[0,9] ∧ F_2∈[0,9] →*discard*. The third error in the above rule sequence is that r_3 is redundant. That is to say, if rule r_3 is removed, the effect of the resulting policy will be unchanged. Such an error is referred to as a compactness error.

3.3 Intra-firewall Anomaly Analysis

In [8], we proposed a firewall rule design method based on multidimensional matrix. By mapping the rules into multidimensional matrix in reverse order, the object rules are non-redundant, conflict-free and completed.

For simplicity, we consider a two-dimensional firewall policy which contains six rules,

r_1:F_1∈[2,6] ∧ F_2∈[4,6] →*accept*,
r_2:F_1∈[8,9] ∧ F_2∈[8,9] →*accept*,
r_3:F_1∈[3,5] ∧ F_2∈[0,5] →*accept*,
r_4:F_1∈[7,8] ∧ F_2∈[0,5] →*discard*,
r_5:F_1∈[6,8] ∧ F_2∈[0,5] →*accept*,
r_6:F_1∈[0,8] ∧ F_2∈[0,9] →*discard*.

After mapping these rules into the multidimensional matrix, we obtain three independent unit spaces: [(3,6)(0,6)],[(8,9)(8,9)],[(2,2)(4,6)], the corresponding firewall rules are

r_1:F_1∈[3,6] ∧ F_2∈[0,6] →*accept*,
r_2:F_1∈[8,9] ∧ F_2∈[8,9] →*accept*,
r_3:F_1∈[2,2] ∧ F_2∈[4,6] →*accept*,
r_4:F_1∈[0,9] ∧ F_2∈[0,9] →*discard*.

It can be seen that the semantics of the object rules are the same as the original policy, while the generated rules r_1,r_2,r_3 are independent of each other without any redundancy and conflicts, and r_4 ensures the completeness.

4 Analysis and Detection of Inter-firewall Anomalies

4.1 Inter-firewall Anomaly Definition

In general, an inter-firewall anomaly may exist if any two firewalls on a network path take different filtering actions on the same traffic. Referring to Figure 1, we assume a traffic flowing from domain D_1 to domain D_2. At any

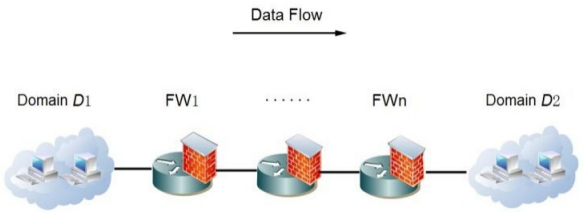


Figure 1: Cascaded firewall isolating domains D_1 and D_2

point on this path in the direction of flow, a preceding firewall is called an upstream firewall, whereas a following firewall is called a downstream firewall. The closest firewall (FW_1) to the flow source domain (D_1) is called the most upstream firewall, while the closest firewall (FW_n) to the flow destination domain (D_2) is called the most downstream firewall.

Even if each firewall policy in the network does not contain the rule anomalies described in Section 3, there could be anomalies between policies of different firewalls. For example, an upstream firewall might block a traffic that is permitted by a downstream firewall or *vice versa*.

As defined in [1], using the network model as shown in Figure 1, for any traffic flowing from domain D_1 to domain D_2, an anomaly exists if one of the following conditions holds:

1) The most downstream firewall accepts a traffic that is blocked by any of the upstream firewalls;

2) The most upstream firewall permits a traffic that is blocked by any of the downstream firewalls;

3) A downstream firewall denies a traffic that is already blocked by the most upstream firewall.

At the same time, all upstream firewalls should permit any traffic that is permitted by the most downstream firewall in order that the flow can reach the destination.

4.2 Inter-firewall Anomaly Classification

In the cascaded firewall network shown in Figure 1, it is assumed that there are no anomalies in the intra-firewall. According to the definition of distributed firewall policy anomaly, if the downstream firewall denies the traffic that has been blocked by the most upstream firewall, a redundancy anomaly will occur. However, we note that according to the definition of intra-firewall anomaly analysis, after eliminating the intra-firewall anomaly, all rules are *accept* except for the last default rule. Considering that the firewall policy follows the principle of "reject everything that is not explicitly allowed", and from the perspective of ensuring the integrity of the rule, we accept the redundancy of this rejection rule. Therefore, for the "condition in definition of the distributed firewall policy anomaly: the downstream firewall denies the traffic blocked by the most upstream firewall.", we do not define this condition as a policy anomaly when detecting the inter-firewall anomalies. In other words, we focus on the following two types of policy anomalies:

1) *Shadowing anomaly* (A_{sh}): the shadowing anomaly occurs if the upstream firewall blocks the network traffic accepted by a downstream firewall;

2) *Spuriousness anomaly* (A_{sp}): the spuriousness anomaly occurs if the upstream firewall permits the network traffic denied by a downstream firewall.

4.3 Inter-firewall Anomaly Detection Algorithm

In this section, we firstly define the related concepts of our approach, and then introduce the algorithm for detecting Inter-firewall anomalies. Table 1 lists the notations used in this article. Taking the distributed firewall network shown in Figure 1 as an example, there are n cascaded firewalls between domains D_1 and D_2, named FW_1,FW_2,...,FW_n respectively. It is assumed that these n firewalls in the network have been processed by the rule mapping method based on multidimensional matrix [8], which means that the rules in each firewall are independent of each other and have decision *accept*, except for the last default *discard* rule. For simplicity, we use blank strips to describe the "*accept*" firewall rule, as shown in Figure 2 and Figure 3. According to the classification of inter-firewall anomalies, in the first case, if an upstream firewall blocks the traffic allowed by the most downstream firewall FW_n, a shadowing anomaly A_{sh} will occur. In another case, a spuriousness anomaly A_{sp} occurs when the most upstream firewall FW_1 allows traffic discarded by a downstream firewall.

Table 1: Notations used in this article

Notation	Paraphrase
F_i	The i^{th} dimension
$D(F_i)$	Domain of F_i
FW_i	The i^{th} Firewall
FW_u	The most upstream firewall
FW_d	The most downstream firewall
R	Firewall rule
US	Unit space
M_k	A k-dimensional matrix
FDM	Firewall design matrix model
A_{sh}	Shadowing anomaly
A_{sp}	Spuriousness anomaly

4.3.1 Shadowing Anomaly Detection

Suppose there are four cascaded firewalls in the network path, named FW_1,FW_2,FW_3 and FW_4 respectively, and

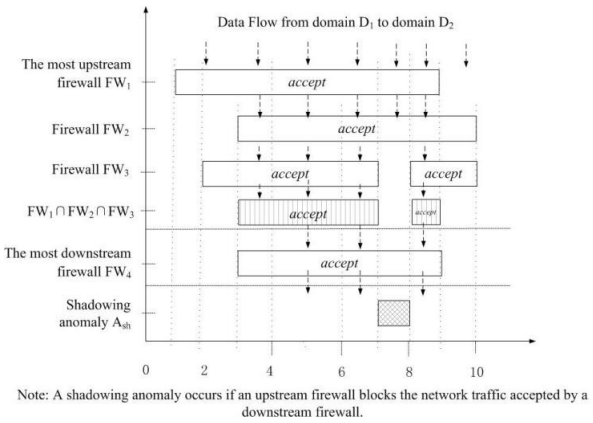


Figure 2: Detecting shadowing anomaly

the blank strip represents the corresponding area of the firewall accept rules, as shown in Figure 2. According to the definition of A_{sh}, shadowing anomaly detection is equivalent to locating the strip that filled with crossing lines. In this case, we first calculate the overlapping area of all the three upstream firewalls accepting rules, denoted as $FW_1 \wedge FW_2 \wedge FW_3$; Then we change the decision of rule in this area to *discard* and map it to the most downstream firewall FW_4; Finally, we obtain the shadowing anomaly A_{sh}.

Next, let us take FW_1 and FW_2 as examples to illustrate how to calculate firewall overlap. First, the decision of the strip area [1,9] representing FW_1 is changed to *discard* and mapped to the corresponding area [3,10] of FW_2, as shown in Figure 2. At this time, the uncovered area of FW_2 is [9,10]; Then we change the decision of these rules to *discard* and map them again to the original area [3,10] of FW_2, where the uncovered area [3,9] of FW_2 is the overlap of FW_1 and FW_2.

According to this method, the overlapping areas of FW_1,FW_2 and FW_3 can be calculated as [3,7] and [8,9]. Finally, the decision of these rules in the overlapping area is changed to *discard* and mapped to the most downstream firewall FW_4, where the uncovered area [7,8] of FW_4 is the shadowing anomaly A_{sh} we seek. This means that the most downstream firewall FW_4 allows packets in [7,8], which are blocked by upstream firewalls, that means a shadowing anomaly occurs in this area. In addition, based on this algorithm, we can obtain the shadowing anomaly between any two firewalls on the distributed firewall network path.

Take any network path in the distributed firewall network, assuming that there are n firewalls FW_1,FW_2,...,FW_n in the network path, where FW_1 is the most upstream firewall, FW_n is the most downstream firewall, and FW_i is the upstream firewall of $FW_j (i < j)$.

Our algorithm includes the following three steps: (1) map all upstream firewall rules of the most downstream firewall into a multidimensional matrix to form a set of independent unit spaces. (2) calculate the unit space overlap of all upstream firewalls. (3) generate the corresponding firewall rules from the overlapping unit space, change the rule decision to *discard* and map it to the multidimensional matrix corresponding to the most downstream firewall. In this case, the unit space area that uncovered by the most downstream firewall is called the Shadowing Anomaly A_{sh}.

1) Rule mapping and rule generating
   According to the rule mapping idea of FDM method, any rule with the form $F_1 \in D(F_1) \wedge \ldots \wedge F_k \in D(F_k) \rightarrow decision$ can be mapped to k-dimensional matrix M_k. In the mapping process, the area with *accept* decision is represented by independent unit spaces US (k-dimensional matrix unit). In order to generate the corresponding firewall rules from the unit spaces, all unit spaces are arranged in descending order according to their area size, and the corresponding firewall rules are respectively generated according to the sorted unit spaces.

2) Calculating the overlapping unit spaces
   For the two unit spaces US_i and US_j, we first generate the corresponding rule set R_i based on US_i, where the rule decision of R_i is *accept*; Then change their decision to *discard* and map it to US_j, at this time, the overlapping area of US_i and US_j can be covered by rule R_i, and then we take the uncovered area of US_j to generate the corresponding rule $R_{j-i \wedge j}$, change its decision to *discard* and map it to the original US_j again, here the unit space not covered in US_j is the overlapping part of US_i and US_j, which is recorded as $US_{i \wedge j}$.

Referring to the above descriptions (1) and (2), the specific process of detecting inter-firewall shadowing anomalies is described in Algorithm 1. The input of the main algorithm is n firewalls FW_1,...,FW_i,FW_j,...,FW_n in a network path, in which FW_i is the upstream firewall of $FW_j (i < j)$, the algorithm output is the Shadowing anomaly (A_{sh}).

4.3.2 Spuriousness Anomaly Detection

As shown in Figure 3, the blank strip represents the area corresponding to the firewall acceptance rule. Accordingly, spuriousness anomaly detection is equivalent to locating the shaded area filled with cross lines. We first calculate the overlapping area of all downstream firewalls, namely $FW_2 \wedge FW_3 \wedge FW_4$; Then we change the decision of the rule in this area to *discard* and map it to the most upstream firewall FW_1. The final area is the spuriousness anomaly A_{sp}.

Specifically, we first change the decision of the blank strip representing FW_2 to *discard* and map it to FW_3, as shown in Figure 3. At this time, the uncovered area of FW_3 is [8,10], we change the decision of this rule to *discard* and map it to the original area [3,10] of FW_3, here the uncovered area [3,8] of FW_3 is the overlapping part of FW_2

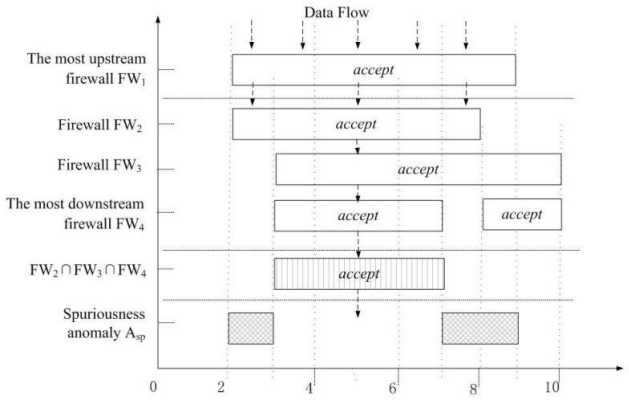
Algorithm 1 Inter-firewall Shadowing Anomaly Detection

1: Begin
2: map $FW_1,\ldots,FW_n$ into M_k to form a set of unit spaces $US_1,\ldots,US_n$.
3: **for** i:=1 to (n-2) **do**
4: $US_{i+1} \leftarrow$ ***Overlap***(US_i, US_{i+1}).
5: **end for**
6: generate rules R_{n-1} from US_{n-1}.
7: change the decision of R_{n-1} to *discard* and map them into US_n.
8: return the uncovered unit spaces in US_n, denoted as A_{sh}.
9: ***Overlap***(US_i, US_j)
10: generate rules R_i from US_i.
11: change the decision of R_i to *discard* and map them into US_j.
12: generate rules $R_{j-i\wedge j}$ from the uncovered unit space in US_j.
13: change the decision of $R_{j-i\wedge j}$ to *discard* and map them into US_j.
14: record the uncovered unit spaces in US_j, which is the overlap of US_i and US_j, denoted as $US_{i\wedge j}$.
15: return $US_{i\wedge j}$.
16: End

and FW_3; Then, we change the decision of rule in [3,8] to *discard* and map it to FW_4, and obtain the overlapping area [3,7] of FW_2, FW_3 and FW_4. Finally, the decision of the overlapping area is changed to *discard* and mapped to the most upstream firewall FW_1, where the uncovered areas [2,3],[7,9] of FW_1 is the desired spuriousness anomaly A_{sp}. This means that the most upstream firewall FW_1 permits packets in [2,3] and [7,9], which are blocked by the downstream firewall, thus spuriousness anomalies occur in this area. In addition, based on this algorithm, we can locate the spuriousness anomalies between any two firewalls on the network path of the distributed firewall environment.

The algorithm process includes the following three steps: (1) all downstream firewall rules of the most upstream firewall are mapped by FDM method respectively to obtain a series of independent unit spaces; (2) calculate the overlap of all unit spaces corresponding to the downstream firewall rules; (3) generate the corresponding firewall rules from the overlapping area, change the rule decision to *discard* and map them to the multidimensional matrix corresponding to the most upstream firewall, in which the uncovered unit space area of the most upstream firewall is the Spuriousness Anomaly A_{sp}. The input of the main algorithm is n firewalls $FW_1,\ldots,FW_i,FW_j,\ldots,FW_n$ in a network path, in which FW_i is the upstream firewall of $FW_j (i<j)$, the algorithm output is the Spuriousness anomaly (A_{sp}).

The execution process of function $Overlap(US_i, US_{i+1})$ is the same as that of Algorithm 1. According to the idea of our inter-firewall anomaly detection algorithm, the time



Note: A spuriousness anomaly occurs if an upstream firewall permits the network traffic denied by a downstream firewall.

Figure 3: Detecting Spuriousness anomaly

Algorithm 2 Inter-firewall Spuriousness Anomaly Detection

1: Begin
2: map $FW_1,\ldots,FW_n$ into M_k to form a set of unit spaces $US_1,\ldots,US_n$.
3: **for** i:=2 to (n-1) **do**
4: $US_{i+1} \leftarrow$ ***Overlap***(US_i, US_{i+1}).
5: **end for**
6: generate rules R_n from US_n.
7: change the decision of R_n to *discard* and map them into US_1.
8: return the uncovered unit spaces in US_1, denoted as A_{sp}.
9: End

complexity of the algorithm mainly depends on the number of firewalls on the network path of the distributed firewall and the execution efficiency of the FDM method. In [8], the time complexity of FDM method is analyzed in detail, which is $O(k^2n^2)$, where n and k are the number and dimension of firewall rules respectively. From the perspective of detection process, one of the main characteristics of this method is that all firewalls in the network path can be regarded as a whole and can comprehensively discover the anomalies caused by multiple rules, it is no longer limited to the traditional inter-firewall anomalies detection algorithm, which can only discover the anomalies between any two rules [1].

5 Performance Analysis and Simulation Results

5.1 Time Complexity Analysis

Suppose that there are t firewalls on the network path from the most upstream firewall FW_1 to the most downstream firewall FW_t. For simplicity, suppose the rules number of each firewall is n, then for Step (1) of Algo-

rithm 1, the time complexity of FDM mapping for t firewalls is $O(tk^2n^2)$. For Step (2), to calculate the overlap of all unit spaces of the upstream firewalls, the execution time of this step is mainly consumed in the circular execution function $Overlap(US_i, US_j)$, while there are two mapping operations during each round of function execution. The first mapping is to generate the rule set R_i from US_i, change its decision value to *discard*, and then overwrite and map it to US_j. Since the number of unit spaces obtained by FDM is generally not more than that of the original rules, the number of unit spaces contained in the t unit space sets is also not more than n. Considering the worst case, each unit space in US_i is included in that of US_j, when mapping each rule corresponding to US_i into a multidimensional matrix, an existing unit space in the multidimensional matrix will be divided into $2k$ sub-unit spaces. Therefore, the time required for the first mapping is:

$$\begin{aligned} & ck\sum_{i=1}^{n}(n-i+2ki) = ck\sum_{i=1}^{n}[n+(2k-1)i] \\ & = ck[n^2+(2k-1)\tfrac{n(n+1)}{2}] \\ & \le ckn^2+ck^2n(n+1) = ck(k+1)n^2+ck^2n \end{aligned} \tag{1}$$

At this time, a maximum of $2kn$ unit spaces in the multidimensional matrix are not covered. Let us continue to consider the second mapping operation, which includes the process of generating the corresponding rule set from the uncovered unit spaces $US_{j-i\wedge j}$ in US_j, changing the rule decision to *discard*, and mapping them into the original US_j. Different from the last mapping operation, there are $2kn$ rules that need to be mapped to the multidimensional space in turn. The time required is:

$$\begin{aligned} & ck\sum_{i=1}^{2kn}(n-i+2ki) = ck\sum_{i=1}^{2kn}[n+(2k-1)i] \\ & = ck[2kn^2+(2k-1)\tfrac{2kn(2kn+1)}{2}] \\ & \le 2ck^2n^2+ck^2n(n+1) = 4ck^3n^2+2ck^3n \end{aligned} \tag{2}$$

Considering the worst case, each unit space in US_i is included in US_j, so the number of unit spaces overlapped by US_i and US_j is exactly n, which means the overlapping $US_{i\wedge j}$ of US_i and US_j contains n unit spaces. The algorithm has t-2 cycles, so the time complexity of the algorithm is $O(tk^3n^2)$, where t is the total number of firewalls, k is the rule dimension and n is the number of rules.

For Step (3), a corresponding rule is generated from the overlapping area of $US_1,\ldots,US_{n-1}$, change the rules decision to *discard*, and map it to the multidimensional matrix space corresponding to the most downstream firewall. This step is equivalent to performing an FDM mapping operation. As mentioned earlier, in the worst case, the overlap of $US_1,\ldots,US_{n-1}$ has n unit spaces corresponding to n rules, so the time complexity of this mapping operation is $O(k^2n^2)$. Based on the above description, the total time of the algorithm is the sum of these three steps. Therefore, the worst-case time complexity of the algorithm is $O(T_{worst})$=$O(tk^3n^2)$.

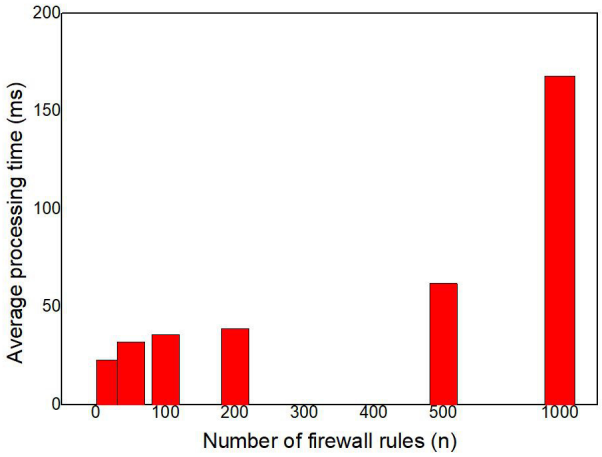


Figure 4: Intra-firewall: The average processing time for eliminating anomalies.

5.2 Experimental Results

In order to test the effectiveness of our method, we refer to the two virtual firewall policies given in [22], and use our algorithm to analyze the policy anomalies between the two firewalls. The specific configurations of the two firewalls are shown in Table 2 and Table 3. The upstream firewall FW_u contains ten rules and the downstream firewall FW_d contains seven rules.

Firstly, the intra-firewall anomaly analysis method is used to eliminate the intra-firewall anomalies in the upstream firewall FW_u and the downstream firewall FW_d respectively; Then, algorithm 1 and algorithm 2 are used to discover the shadowing anomaly and the spuriousness anomaly. The algorithms detects ten anomalies, including four shadowing anomalies and six spuriousness anomalies, which are consistent with the detection results in [22] and the actual situation. These six spuriousness anomalies are shown in Table 4.

Specifically, we first map FW_u and FW_d to M_k to form a set of unit spaces US_u and US_d respectively; Then we generate rules R_d from US_d, change the decision of R_d to *discard*, and map it to US_u; Finally, the uncovered unit spaces in the US_u are the spuriousness anomaly A_{sp} we desired.

For example, data packet "source IP ='B', destination IP ='M', source port ='*', destination port ='25', protocol ='TCP'" matches rule 3. Obviously, the packet is allowed to pass through the most upstream firewall FW_u, and blocked by the most downstream firewall FW_d. As mentioned earlier, if the upstream firewall permits network traffic denied by the downstream firewall, a spuriousness anomaly will occur. Therefore, rule 3 conforms to the definition of spuriousness anomaly.

In order to evaluate the efficiency of the proposed algorithm, we use *Classbench* [20] to generate six groups of firewall policies, each of which contains 20, 50, 100, 200, 500 and 1000 rules respectively. Each group con-

Table 2: The upstream firewall FW_u policy

	SourceIP	DestIP	SourcePort	DestPort	Prot	Action
1	A,B,C	H	*	80	TCP	*discard*
2	B,C	M,N	*	23,25	TCP	*discard*
3	*	M,N	*	*	TCP	*accept*
4	*	*	*	*	TCP	*discard*
5	C,D,E	H,K	*	53	UDP	*discard*
6	C,D,E	*	*	53	UDP	*accept*
7	*	*	*	*	UDP	*discard*

Table 3: The downstream firewall FW_d policy

	SourceIP	DestIP	SourcePort	DestPort	Prot	Action
1	D,E,F	O	*	80	TCP	*accept*
2	B	O	*	80	TCP	*accept*
3	B	M,N	*	25	TCP	*accept*
4	B	M,N	*	23	TCP	*accept*
5	E,F	*	*	139	UDP	*accept*
6	F	H	*	53	UDP	*accept*

Table 4: Spuriousness anomalies detection result

	SourceIP	DestIP	SourcePort	DestPort	Prot	Action
1	A,B,C	H	*	80	TCP	*discard*
2	A,B	M,N	*	23,25	TCP	*accept*
3	*	M,N	*	23,25	TCP	*discard*
4	A,C	O	*	*	TCP	*discard*
5	*	O	*	80	TCP	*accept*
6	*	*	*	*	TCP	*discard*
7	C,D,E	H	*	53	UDP	*discard*
8	*	H	*	53	UDP	*accept*
9	E,F	*	*	139	UDP	*accept*
10	*	*	*	*	UDP	*discard*

Table 5: The time(ms) of anomalies detection

Method	Anomalies	Link100	Link200	Link300	Link400	Link500
Method-W	A_{sh}	279.24	337.30	351.25	362.62	400.46
	A_{sp}	26.61	32.23	35.02	50.45	54.20
Method-C	A_{sh}	212.45	264.05	300.26	335.80	371.45
	A_{sp}	14.24	26.05	21.65	35.06	37.09

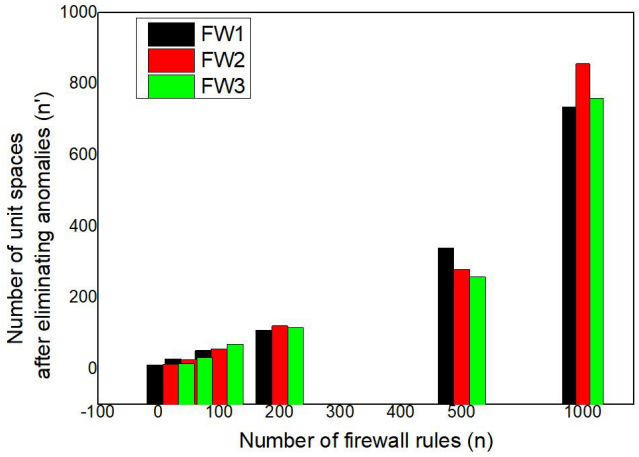


Figure 5: Intra-firewall: The number of rules after eliminating anomalies.

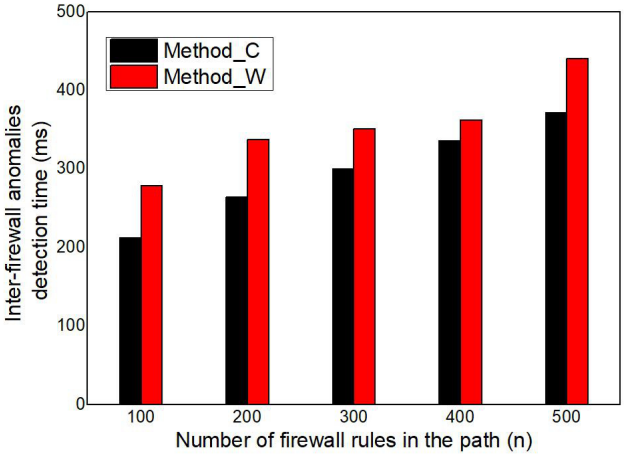


Figure 6: Processing time of detecting the inter-firewall anomalies

tains three firewall policies on average, named FW_1, FW_2 and FW_3. These algorithms were implemented in Java JDK 1.6, and we conducted our experiments on a desktop PC running Windows 7 with 4.0G memory and Intel(R) Core(TM) Processor of 2.60GHz.

We first eliminate the intra-firewall anomalies, the average running time of the program is shown in Figure 4. with the increase of the number of rules in firewall policy, the time required to eliminate policy anomalies increases accordingly. For example, when the number of firewall rules is 500, the system processing time is $65ms$, while when the number of firewall rules reaches 1000, the system only needs about $160ms$. It can be seen that the execution time of the system roughly conforms to the quadratic function curve, which also verifies the time complexity $O(k^2n^2)$ of FDM algorithm, and the execution efficiency of the method is high. Figure 5 shows the amount of unit space in each group of firewalls after eliminating policy anomalies. The result also verify that the number of unit spaces is less than the number of original rules.

In order to further evaluate the time performance of the inter-firewall spuriousness anomaly detection algorithm, we designed five different network paths to obtain the average processing time of the algorithm. For convenience, each path contains four firewalls with the same number of rules, expressed as FW_1, FW_2, FW_3, FW_4 from the most upstream firewall to the most downstream firewall. The number of firewall rules in these five network paths is 100, 200, 300, 400 and 500, respectively. Accordingly, we record them as path 100, path 200, path 300, path 400, and path 500. For example, path 100 means that there are four firewalls in the network path, and each firewall has 100 rules. The names of other paths follow the same principle. We execute algorithm 1 and algorithm 2 on the five paths respectively, the required running time of shadowing anomalies and spuriousness anomalies detected are shown in Table 5.

Figure 6 shows the time taken to detect the inter-firewall anomaly for each path using the *Method-C* herein and the *Method-W* described in [22], respectively. With the increase of the number of firewall rules in the path, the time required to detect firewall anomalies also increases. For a network path with four firewalls, especially when the number of rules in each firewall reaches 500, the method in this paper can detect the shadowing anomalies in only $370ms$. From the comparison results, it can be seen that our algorithm has higher execution efficiency.

6 Conclusions

This paper presents a method for detecting and eliminating the intra-firewall anomalies. This method can completely discover the anomalies while maintaining the consistency, compactness and completeness of the original firewall rules. Then the definition and classification of inter-firewall anomalies are discussed, and an approach of inter-firewall anomalies detection based on rule space comparison is proposed. Theoretical analysis and simulation results show that this method can detect shadowing anomalies and spuriousness anomalies accurately and efficiently.

Acknowledgments

This study was supported by Research Foundation of the Education Department of Hunan Province (No. 21C1589), the Doctoral Scientific Research Project of Changsha Social Work College(2020JB32), and the National Natural Science Foundation of China (61877059). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] E. S. Al-Shaer and H. H. Hamed, "Discovery of policy anomalies in distributed firewalls," in *Proceedings of The IEEE INFOCOM (INFOCOM'04)*, pp. 2605–2616, Hong Kong, China, March 2004.

[2] J. G. Alfaro, F. Cuppens, and N.Cupperns-Boulahia, "Analysis of policy anomalies on distributed network security setups," in *Proceedings of The European Symposium on Research in Computer Security (ESORICS'06)*, pp. 496–511, Hamburg, Germany, Sept. 2006.

[3] F. Baboescu, S. Singh, and G. Varghese, "Packet classification for core routers: is there an alternative to cams?," in *Proceedings of The IEEE INFOCOM (INFOCOM'03)*, pp. 53–63, San Francisco, USA, March 2003.

[4] D. Bringhenti, G. Marchetto, and R. Sisto, "Automated optimal firewall orchestration and configuration in virtualized networks," in *Proceedings of The IEEE/IFIP Network Operations and Management Symposium (NOMS'20)*, pp. 1–7, Budapest, Hungary, April 2020.

[5] C. S. Chao, "A flexible and feasible anomaly diagnosis system for internet firewall rules," in *Symposium of The Asia-Pacific Network Operations and Management (APNOMS'11)*, pp. 1–8, Taipei, Taiwan, Sept. 2011.

[6] F. Chen, A. X. Liu, and J. H. Hwang, "First step towards automatic correction of firewall policy faults," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 7, no. 2, pp. 1–15, 2012.

[7] Y. X. Cheng, H. J. Yao, Y. Wang, Y. Xiang, and H. P. Li, "Protecting vnf services with smart online behavior anomaly detection method," *Future Generation Computer Systems*, vol. 95, pp. 265–276, 2019.

[8] Y. Z. Cheng, W. P. Wang, G. Y. Min, and J. X. Wang, "A new approach to designing firewall based on multidimensional matrix," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 12, pp. 3075–3088, 2015.

[9] Y. Z. Cheng, W. P. Wang, J. X. Wang, and H. D. Wang, "Fpc: a new approach to firewall policies compression," *Tsinghua Science and Technology*, vol. 24, no. 1, pp. 65–76, 2019.

[10] M. G. Gouda and A. X. Liu, "Structured firewall design," *Computer Networks*, vol. 51, no. 4, pp. 1106–1120, 2007.

[11] H. H. Hamed and E. Al-Shaer, "Taxonomy of conflicts in network security policies," *IEEE Communications Magazine*, vol. 44, no. 3, pp. 134–141, 2006.

[12] H. X. Hu, G. Ahn, and K. Kulkarni, "Detecting and resolving firewall policy anomalies," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 3, pp. 318–331, 2012.

[13] J. H. Hwang, T. Xie, F. Chen, and A. X. Liu, "Systematic structural testing of firewall policies," *IEEE Transactions on Network and Service Management*, vol. 9, no. 1, pp. 1–11, 2012.

[14] M. S. Hwang and W. P. Yang, "Controlling access in large partially-ordered hierarchies using cryptographic key," *The Journal of Systems and Software*, vol. 67, no. 2, pp. 99–107, 2003.

[15] T. Kim, T. Kwon, and J. Lee, "F/wvis: hierarchical visual approach for effective optimization of firewall policy," *IEEE Access*, vol. 9, pp. 105989 – 106004, 2021.

[16] L. H. Liu, Z. J. Cao, and M. Chong, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[17] B. E. Logan and G. G. Xie, "Automating distributed firewalls: a case for software defined tactical networks," in *Proceedings of The IEEE Military Communications Conference (MILCOM'19)*, pp. 1–6, VA, USA, Nov. 2019.

[18] L. Maccari and R. L. Cigno, "Privacy in the pervasive era: a distributed firewall approach," in *Proceedings of The 9th Annual Conference on Wireless On-Demand Network Systems and Services (WONS'12)*, pp. 23–26, Courmayeur, Italy, Jan. 2012.

[19] S. R. Pedditi, D. Zhang, and C. Wang, "Fiep: an initial design of a firewall information exchange protocol," in *Proceedings of The 14th International Conference on Information Reuse and Integration (IRI'13)*, pp. 1–5, San Francisco, USA, Aug. 2013.

[20] D. E. Taylor and J. S. Turner, "Classbench: a packet classification benchmark," *IEEE/ACM Transactions on Networking*, vol. 15, no. 3, pp. 499–511, 2007.

[21] C. Togay, A. Kasif, C. Catal, and B. Tekinerdogan, "A firewall policy anomaly detection framework for reliable network security," *IEEE Transactions on Reliability*, vol. 99, pp. 1–9, 2021.

[22] W. P. Wang, W. H. Chen, W. P. Zhu, H. P. Chen, and J. Yang, "Analysis of distributed firewall policy configuration mistakes and their detection," *Journal of University of Chinese Academy of Science(in Chinese)*, vol. 24, no. 2, pp. 257–265, 2007.

[23] S. Yingchareonthawornchai, J. Daly, A. X. Liu, and E. Torng, "A sorted-partitioning approach to fast and scalable dynamic packet classification," *IEEE/ACM Transactions on Networking*, vol. 26, no. 4, pp. 1907–1920, 2018.

[24] L. H. Yuan, J. N. Mai, Z. D. Su, H. Chen, C. N. Chuah, and P. Mohapatra, "Fireman: a toolkit for firewall modeling and analysis," in *Proceedings of The IEEE Symposium on Security and Privacy (SSP'06)*, pp. 199–213, California, USA, Jan. 2006.

Biography

Yu-zhu Cheng received the B.S. degree from Hunan university of science and technology in 2002 and the M.S. degree in software engineering from Hunan University in 2005, and the Ph.D. degree in computer science and technology from Central South University

in 2018. Currently, he is an associate professor at Changsha Social work College. His current research interests include data privacy and information security. He has published more than 30 papers in referred journals and conference proceedings, he is a member of IEEE.

Qiu-ying Shi received the B.S. degree from Hunan Normal University and the M.S. degree in computer science and technology from Central South University. Her research interests include cyber security and pattern recognition.

An Adaptive Speech Homomorphic Encryption Scheme Based on Energy in Cloud Storage

Qiu-Yu Zhang and Yu-Jiao Ba
(Corresponding author: Qiu-Yu Zhang)

School of Computer and communication, Lanzhou University of Technology
No. 287, Lan-Gong-Ping Road, Lanzhou 730050, China
Email: zhangqylz@163.com, bayujiaolut@163.com
(Received Oct. 26, 2021; Revised and Accepted Apr. 15, 2022; First Online Apr. 23, 2022)

Abstract

Aiming at the problems of the traditional speech encryption scheme in cloud storage, such as security risks, excessive communication consumption, low robustness to resist multiple types of attacks, and low efficiency of the speech homomorphic encryption scheme, an adaptive speech homomorphic encryption scheme based on energy in cloud storage was proposed. Firstly, by comparing the threshold of speech energy, the improved Adaboost algorithm is used to design an adaptive classifier. Then, the speech data is divided into the sound and silent parts according to the energy threshold. Secondly, the BGV homomorphic encryption algorithm is used to encrypt the sound part of the information. Then, the Paillier homomorphic encryption algorithm is used to encrypt the silent part of the information. Finally, the two parts of ciphertext are combined to realize ciphertext domain operation and adaptive decryption. The experimental analysis shows that the proposed scheme has good encryption and decryption efficiency and low ciphertext expansion and can resist various attacks (including statistical, entropy, and chosen-plaintext attacks).

Keywords: Adaptive Classifier; BGV Homomorphic Encryption; Homomorphic Encryption; Paillier Homomorphic Encryption

1 Introduction

With the rapid development of the cloud storage and Internet technology, more and more users choose to store multimedia data uploaded to the cloud. Cloud storage separates the ownership and management rights of data [20,26], which makes the security of multimedia data and the protection of personal privacy in cloud storage attract people's attention [4, 10]. Therefore, ensuring the security of speech content has become an important research issue. As a standard and an effective technology to protect the security of digital multimedia information content, speech encryption plays an important role in the applications of speech retrieval [18].

At present, common speech encryption methods include homomorphic encryption [6, 15, 16], chaotic mapping encryption [1] (including Lorenz mapping, Logistic mapping, Henon mapping, etc.), scrambling encryption, RSA, AES, etc. Since homomorphic encryption can not only protect data privacy, but also allow the operation of encrypted data (such as simple addition, subtraction and multiplication). It can support the extraction of effective features from encrypted data [20]. By analyzing the full homomorphic encryption scheme [17] and the applications of homomorphic encryption, homomorphic encryption is more and more widely used in the field of data encryption [9] and multimedia data encryption (such as image data [25], speech data [6, 15, 16, 20], etc.). Speech homomorphic encryption has become one of the key components of secure speech storage in public cloud computing.

Adaptive control [13] can automatically adjust the processing method, processing sequence, processing parameters, boundary conditions or constraints according to the data characteristics of the processed data to adapt to the statistical distribution characteristics and structural characteristics of the processed data, so as to obtain the best processing effect and realize random optimization. When parameters are estimated from massive speech data according to the optimal scheme, adaptive control can abandon the local optimal in the solution space and tends to the global optimal [8], which is more suitable for massive speech classification in the cloud environment. In recent years, the multimedia data combined with adaptive algorithms, encryption algorithms and other methods have made great achievements in the fields of data encryption [11], image encryption, speech encryption [23] and video encryption. Adaptive technology has also made great achievements in the research of speech processing fields such as semantic analysis, speech retrieval [3], audio watermarking, etc. Adaptive encryption methods can generate speech residual similar to any other speech signal, which can further protect the security of speech data [7, 14, 21]. Therefore, for the practical

applications such as speech data security and ciphertext speech retrieval in cloud storage environment, it is of great significance to research adaptive homomorphic encryption methods suitable for the characteristics of speech signals.

To solve the above problems, an adaptive speech homomorphic encryption scheme based on energy in cloud storage is proposed to ensure the security of cloud data and adaptive encryption for speech signal characteristics. The main contributions of this work are as follows:

1) In order to improve the efficiency of the encryption scheme, the parameters are estimated by comparing the threshold of speech energy, and an adaptive classifier is designed to reduce the complexity of low-energy data encryption and strengthen the robustness against various types of attacks (including statistical attack, entropy attack, and chosen-plaintext attack).

2) By using adaptive parameters to classify speech data based on energy, the original speech data is divided into multiple data blocks according to the energy threshold, and the data blocks are numbered. The strong correlation between adjacent speech blocks is reduced by different homomorphic encryption for the data blocks of the sound and silent part respectively.

3) Parallel adaptive encryption and decryption. Different homomorphic encryption is carried out for the speech data of the sound and silent part after the adaptive selection. Similarly, the parallel decryption is carried out after the adaptive ciphertext differentiation, which minimizes the amount of computation while maintaining a certain computational complexity and improves the encryption efficiency of the encryption scheme.

The rest of the paper is arranged as follows. Section 2 analyzes relevant research work in detail. Section 3 gives the system model of the encryption scheme, and describes the adaptive speech homomorphic encryption algorithm and its processing process in detail. Section 4 analyzes the encryption performance of the encryption scheme. Section 5 gives the experimental results and the performance analysis compared with existing schemes. Finally, we conclude our work in Section 6.

2 Related work

Speech encryption is one of the key steps in ciphertext speech retrieval. In recent years, while exploring homomorphic speech encryption [6, 15, 16], chaotic speech encryption [18] and other encryption schemes, many scholars have proposed adaptive encryption technology for speech data [7, 21] to protect the privacy and security of data in the cloud environment [19]. At present, the combination of multimedia data and adaptive methods has made considerable achievements in image and speech encryption [25]. Adaptive speech encryption is robust to multiple types of attackers (including ciphertext attackers and plaintext attackers), it is not an encryption algorithm, but a combination of speech signal processing and multiple encryption technologies.

Aiming at the security of speech data in cloud storage, Shi [16] proposed a digital speech encryption scheme based on homomorphic encryption by using the symmetric key cryptosystem (MORE-method) with probability statistics and complete homomorphism characteristics to encrypt speech signals, but this scheme has a large ciphertext expansion and cannot resist statistical analysis estimation. In order to solve the above problems, Shi [15] improved the scheme in 2019 and proposed a probability statistics addition homomorphic encryption scheme with small expansion of ciphertext. This scheme limits the expansion of ciphertext data and resists statistical analysis attacks. Imran [6] proposed the El-Gamal speech homomorphic encryption scheme. The security of this scheme is based on computing discrete logarithmic moduli of large prime numbers, which would take thousands of years for attackers to crack. In order to solve the problems of the above schemes, this paper will use the existing efficient homomorphic encryption scheme to encrypt and decrypt the speech data. BGV (Brakerski-Gentry-Vaikuntanathan) homomorphic encryption [2,5] is the most efficient scheme among the current mainstream homomorphic encryption algorithms. Using homomorphic encryption can realize the operation of addition, subtraction and multiplication in the encryption domain, and can further realize the feature extraction operation in the ciphertext domain. Paillier [12] algorithm is the most commonly used and practical additive homomorphic encryption algorithm, which has been applied in many application scenarios. In this paper, BGV algorithm and Paillier algorithm are used for encryption, and the ciphertext can be filtered by a step of multiplication before decryption to support different decryption operations.

In order to further improve the data security in the cloud, Shahadi [14] proposed an adaptive speech encryption method, combining the biggest advantages of cryptography and steganography, and adapting the wavelet coefficients of encrypted speech to any other speech signal coefficients. Neither send the encrypted content of the secret speech nor extend the bandwidth of the transmission message. The ciphertext speech retrieved by the scheme shows high speech quality and is robust to both ciphertext-only and plaintext attacks. Jahanshahi [7] designed a robust adaptive control scheme to encrypt speech. The adaptive mechanism was used to estimate the unknown parameters of the system, and the output of the proposed adaptive mechanism was used in the control scheme to achieve a fractional order system of speech encryption. But its efficiency limits the application of the system to a great extent. In order to improve the efficiency of adaptive encryption, Tutueva [21] proposed a new pseudo-random generation method based on the concept of adaptive symmetric chaotic mapping. Adaptive coefficients combined with chaos-based Pseudo-Random

Number Generator (PRNG) are easier to implement than existing chaos-based improved generators, and chaotic maps with adaptive symmetry are suitable for stream ciphers. However, the various attacks on cryptographic systems based on adaptive mapping are not discussed in this paper, which is not enough to prove their security.

In summary, the existing adaptive encryption schemes and homomorphic encryption schemes are mostly used in image fields, and the existing speech adaptive schemes are mostly combined with chaotic encryption and scrambling encryption. There are relatively few studies on sensitive speech data, and the combination of homomorphic encryption and adaptive control is rarely applied to speech data. To solve these problems, an energy-based adaptive speech homomorphic encryption scheme is proposed, which can implement adaptive selective encryption for speech data, making the encryption more efficient and less data expansion, and having strong robustness to a variety of types of attackers.

3 The Proposed Scheme

3.1 System Model

Figure 1 shows the system model in the proposed scheme. The system model consists of four entities: data owner (DO), cloud server (CS), adaptive classifier (AC), and retrieval user (RU).

As shown in Figure 1, the components of the system model accomplish the following:

Data Owner (DO): DO owns local speech data $S = S_1, S_2, \ldots, S_n$. To ensure the privacy and security of speech data, the speech data is encrypted after adaptive classification, and the ciphertext speech data $C = C_1, C_2, \ldots, C_n$ is obtained. Where n represents the number of speech data. Finally, the generated ciphertext speech data C is outsourced to CS for storage.

Adaptive Classifier (AC): AC is an adaptive classifier for speech data generated by threshold estimation. In order to improve the encryption performance, the original speech is classified into sound data $S' = \{S'_1, S'_2, \ldots, S'_n\}$ and silent data $S'' = \{S''_1, S''_2, \ldots, S''_n\}$ by adaptive selection, and the ciphertext speech data $C' = \{C'_1, C'_2, \ldots, C'_n\}$ and $C'' = \{C''_1, C''_2, \ldots, C''_n\}$ are obtained after parallel encryption. Finally, the ciphertext speech data $C = C_1, C_2, \ldots, Cn$ is generated.

Cloud Server (CS): CS stores ciphertext speech data C uploaded by DO and performs ciphertext calculations on C to obtain the new ciphertext C^*. When receiving RU's search request, CS returns the query result C^* to RU.

Retrieval User (DU): DU decrypts the plaintext speech data by using the key sent by DO after receiving the query result from CS.

3.2 Adaptive Classifier

In the process of processing and analysis, adaptive control automatically adjusts the processing method, processing sequence, processing parameters, boundary conditions or constraints according to the data characteristics of the processed data to adapt to the statistical distribution characteristics and structural characteristics of the processed data, so as to obtain the best processing effect. Adaptive control usually uses the adaptive algorithm to generate online estimation of unknown parameters.

The adaptive boosting (Adaboost) algorithm [24] is improved to combine multiple weak classifiers into a strong classifier in this paper. The principle of adaptive classifier is to adjust its parameters according to some criteria and adaptive algorithm to minimize the cost (objective) function of adaptive classifier and achieve the purpose of optimal equilibrium. Figure 2 shows the adaptive classifier (AC) model designed by Adaboost algorithm in the proposed scheme.

The dotted line in Figure 2 represents the iteration effect of different rounds, and each iteration adds a classification structure. The work of the i-th iteration is as follows:

1) Add weak classifier Y_i and weak classifier weight $\mathrm{Alpha}(i)$;

2) The weak classifier Y_i is trained by data set Data and data weight $\mathrm{W}(i)$, and its classification error rate is obtained, so as to calculate its weak classifier weight $\mathrm{Alpha}(i)$;

3) Combine each trained weak classifier Y_i into an adaptive classifier AC. After the training process of each weak classifier is over, if the final error rate is lower than the set threshold (this paper is set to 3%), then the iteration ends; if the final error rate is higher than the set threshold, then update the data weight to get $\mathrm{W}(i+1)$.

The basic principle of the adaptive classifier designed in this paper is to combine multiple weak classifiers (weak classifiers generally use single-layer decision trees) to make them a strong classifier. The algorithm adopts the idea of iteration. Only one weak classifier is trained for each iteration, and the trained weak classifier will participate in the use of the next iteration. That is, in the i-th iteration, there are a total of i weak classifiers, of which $i-1$ are already trained, and their various parameters are no longer changed. This time the i-th classifier is trained. The relationship of the weak classifier is that the i-th weak classifier is more likely to match the data that the first $i-1$ weak classifier did not match, and the final classification output is the comprehensive classification result of the i classifiers.

There are two kinds of weights in the Adaboost algorithm, one is the weight of the data, and the other is the weight of the weak classifier. Where the weight of the data is mainly used for the weak classifier to find the decision

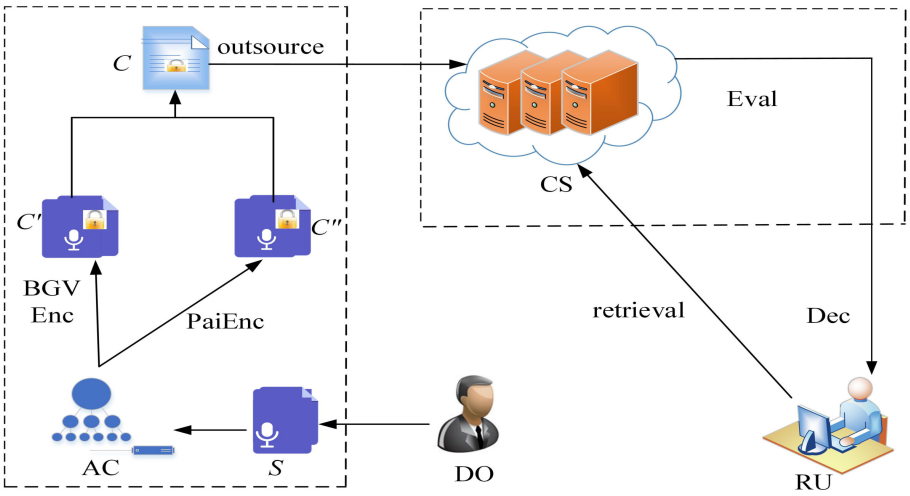


Figure 1: Energy-based adaptive speech homomorphic encryption system model

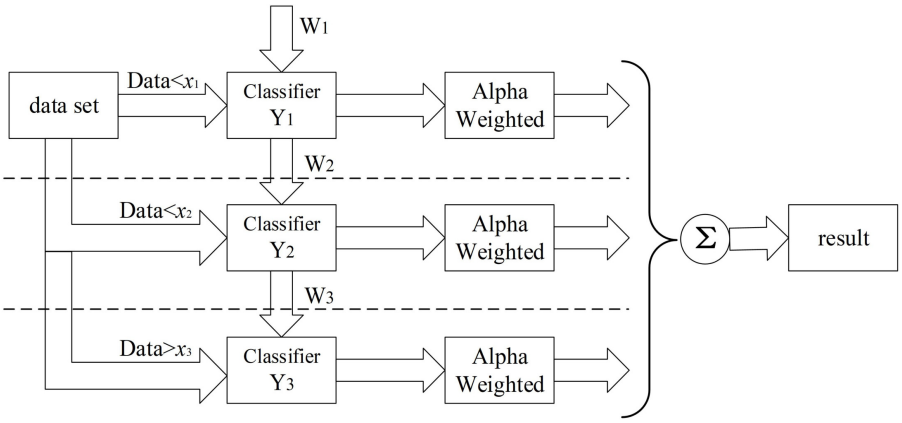


Figure 2: Adaptive classifier (AC) model

point with the smallest classification error. The weight of a weak classifier depends on its error rate. The lower the error rate, the higher the weight. In Adaboost, each weak classifier has its own threshold, and each weak classifier only focuses on a part of the entire dataset, so they must be combined to achieve the final classification.

3.3 Adaptive Homomorphic Encryption

Figure 3 shows the specific processing flow of the energy-based adaptive speech homomorphic encryption scheme. After the preprocessed original speech is classified by the designed adaptive classifier, the homomorphic encryption of different classes of speech data is performed in parallel. When the speech frame data belongs to the -1 category, it is subjected to BGV homomorphic encryption. When the speech frame data belongs to the $+1$ category, it is subjected to Paillier homomorphic encryption.

The specific processing steps are as follows:

Step 1: Pretreatment. Read the original speech data and perform smoothing processing to obtain the value range of the data.

Step 2: Adaptive classification. The speech data is divided into -1 category and $+1$ category through the trained adaptive classifier.

Step 3: Batch packaging. Use the Chinese remainder theorem (CRT) and single instruction multiple data (SIMD) to pack the classified data into a one-dimensional array to implement parallel encryption operations.

Step 4: Adaptive homomorphic encryption. Perform BGV homomorphic encryption on category -1 sound data; perform Paillier homomorphic encryption on category $+1$ silent data.

The security of the BGV homomorphic encryption algorithm is based on the shortest vector problem (SVP). The algorithm establishes a new method to construct a FHE scheme with a fixed circuit depth without Gentry's bootstrapping (able to evaluate circuits of arbitrary polynomial size). The ciphertext multiplication operation will

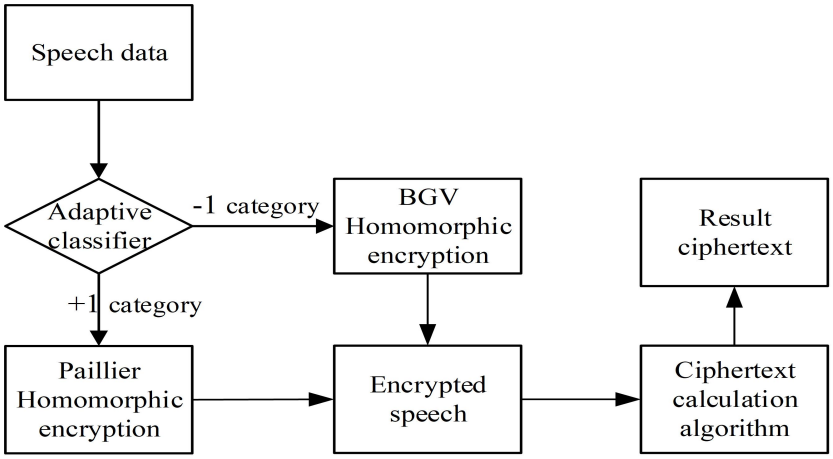


Figure 3: Adaptive speech homomorphic encryption processing flowchart

cause the explosive growth of the ciphertext dimension, resulting in the solution can only perform a constant number of multiplication operations. But the algorithm can use key exchange technology and module exchange technology to solve this problem: the key exchange technology can control the dimensional expansion of the ciphertext vector. After the ciphertext is calculated, the expanded ciphertext dimension is restored to the original ciphertext dimension through key exchange; Modular switching technology can replace the Bootstrapping process in the Gentry scheme to control the noise increase generated by the homomorphic operation of ciphertext. Therefore, after each ciphertext multiplication operation, it is necessary to reduce the dimensionality of the ciphertext through the key exchange technology, and reduce the noise of the ciphertext through the modular exchange technology, so that the next calculation can be continued.

The security of the Paillier homomorphic encryption algorithm is based on the complex remaining difficult problems, and it is a public key encryption algorithm. Before encryption and decryption, public keys n and g that can be used for encryption must be generated. n is the product of two large prime numbers of similar size:$n = p \cdot q$. g is a semi-random number, and its order must be in $Z^*_{n^2}$, that is, the order of g modulo n_2 must be a multiple of n. The public key used for the actual encryption and decryption operation process is (n, g). With the release of the public key, anyone can use the public key to encrypt data and pass the ciphertext to the private key holder.

3.4 Encryption and Decryption Scheme

Figure 4 is the processing flow of this text encryption and decryption scheme, which mainly includes three parts of processing work. First, the adaptive classifier AC is designed, and the classifier model used for homomorphic encryption is trained to perform adaptive homomorphic encryption. Then the data owner DO sends the speech database to AC for adaptive classification, adaptively encrypts the -1 and +1 speech data, stores the encrypted speech in the cloud, and CS performs outsourcing calculation and retrieval. Finally, the authorized user RU decrypts the retrieved speech returned from the cloud to obtain the decrypted speech.

The definitions of symbols used in the proposed scheme are shown in Table 1.

Table 1: Symbol definitions

Symbol	Definitions
$S = S_1, S_2, \ldots, S_n$	Speech data set
$C = C_1, C_2, \ldots, C_n$	Encrypted speech data set
$SK = sk_1, sk_2$	Private key sk_1, sk_2
$PK = pk_1, pk_2$	Public key pk_1, pk_2
$EVK = evk_1, evk_2$	Calculation key evk_1, evk_2
i	Number of iterations
N	Total number of sample data

The proposed adaptive speech homomorphic encryption and decryption scheme consists of 5 algorithms: **Setup**, **GenKey**, **Enc**, **Eval**, **Dec**.

1) Adaptive classifier algorithm. $m_i \leftarrow$**Setup**(S, i). Input speech sample data set $S = S1, S2, \ldots, Sn$ and the number of iterations i, Generate a strong classifier $G(x)$ and output the classification result m_i.

2) Key generation algorithm. $sk, pk, evk \leftarrow$**GenKey** (λ, p, q). This algorithm is a probabilistic key generation algorithm. Enter λ, p, q, and return the private key (sk_1, sk_2), public key (pk_1, pk_2) and ciphertext calculation key (evk_1, evk_2).

3) Speech encryption. $c \leftarrow$**Enc**(pk, m). This algorithm is a probabilistic algorithm. Input the public key pk and the speech data m, and return the ciphertext speech data c.

4) Ciphertext calculation algorithm. $c' \leftarrow$ **Eval** (evk, C, c). Input the ciphertext calculation key evk,

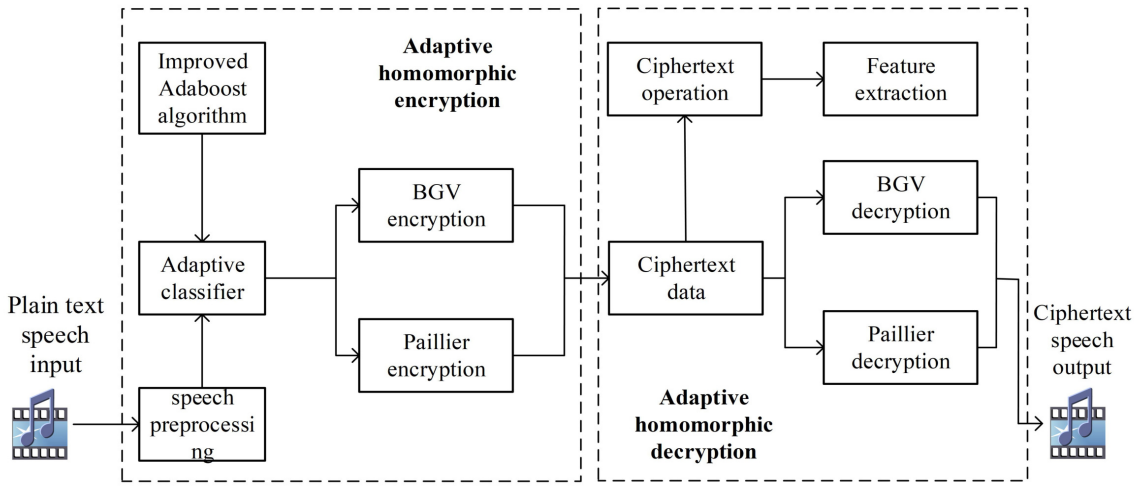


Figure 4: Adaptive speech homomorphic encryption and decryption processing flow

circuit C and ciphertext c, and output the ciphertext calculation result c'.

5) Speech decryption. $m' \leftarrow \mathbf{Dec}(sk, c')$. This algorithm is a deterministic algorithm. Enter the private key sk and the ciphertext calculation result c', and return the decrypted speech m'.

The processing process of the adaptive speech homomorphic encryption and decryption system is as follows:

Step 1: Adaptive classification. The parameter adaptive algorithm is combined with the discrete system to obtain the estimated parameters.

After 3 iterations, the process of implementing adaptive classification is as follows:

1: Initialize the weight distribution of the training data (each sample). Each training sample is initialized with the same weight $W_1 = 1/N$.

2: Perform multiple iterations, $i = 1, 2, 3$, i represents the number of iterations. The adaptive classifier designed in this paper iterates 3 times in total.

1) Use the training sample set with the weight distribution $w_i (i = 1, 2, 3)$ for learning, and get the weak classifier $\mathrm{Y}_i(x)$. The criterion is shown in Equation (1). The error function of the weak classifier is the smallest, that is, the sum of the weights corresponding to the wrong samples is the smallest.

$$\varepsilon_i = \sum_{n=1}^{N} w_n^{(i)} I\left(\mathrm{Y}_i\left(x_n\right) \neq t_n\right) \tag{1}$$

$$\mathrm{Y}_i\left(x\right) : \chi \rightarrow \{-1, +1\} \tag{2}$$

2) Calculate the weight of the weak classifier $\mathrm{Y}_i(x)$, and the weight $w(i)$ indicates the importance of $\mathrm{Y}_i(x)$ in the final classifier.

$$w(i) = \frac{1}{2} \log \frac{1 - \varepsilon_i}{\varepsilon_i} \tag{3}$$

This value increases as ε_i decreases. That is, a classifier with a small error rate is more important in the final classifier.

3) Update the weight distribution of the training sample set. Used in the next iteration. The weight of the misclassified samples will increase, while the weight of the correct score will decrease.

3: After the iteration is completed, the combined weak classifier is the final adaptive classifier AC.

$$\mathrm{AC} = \sum_{i=1}^{3} w(i) \mathrm{Y}_i(x) \tag{4}$$

Step 2: Key generation. This scheme is a multi-key homomorphic encryption system, and the key generation algorithm is composed of two key generation functions.

1: BGV key generation

Randomly select an element on χ as the private key: $sk_1 = s \leftarrow \chi$, take $\mathbf{a} \leftarrow R_q, \mathbf{e} \leftarrow \chi$, and get the public key $pk_1 = \left(\left[-\left(\mathbf{a}s + \mathbf{e} \right) \right]_q, \mathbf{a} \right)$. Thus, the key pair (sk_1, pk_1) is obtained.

2: Paillier key generation

In the case of the same key length, the keys $g = n + 1, \lambda = \varphi(n), \mu = \varphi(n) - 1 \mathrm{mod} n$ can be quickly generated, where $\varphi(n)$ refers to the Euler function, and its value is $(p-1) \times (q-1)$. Get the key pair (sk_2, pk_2), the public key $pk_2 = (n, g)$, and the private key $sk_2 = (\lambda, \mu)$.

Step 3: Speech encryption. Use BGV homomorphic encryption to encrypt the sound part of the speech information; Paillier homomorphic encryption to encrypt the silent part of the speech information.

1: BGV encryption

Encryption algorithm of BGV homomorphic encryption: $c_1 = ([\Delta \cdot m + p[0]\mathbf{u} + \mathbf{e}_1]_q, [p[1]\mathbf{u} + \mathbf{e}_2]_q)$. Get the ciphertext c_1.

2: Paillier encryption

1) The plaintext m is a positive integer greater than or equal to 0 and less than n.
2) Randomly select r to satisfy $0 < r < n$ and $r \in Z_{n^2}^*$ (a sufficient condition is that r and n are relatively prime). $r \in Z_{n^2}^*$ means that r has a multiplicative inverse element in the remainder of n^2. Get the ciphertext $c_2 = g^m r^n \bmod n^2$.

Step 4: Ciphertext calculation. The ciphertext calculation algorithm for homomorphic encryption.

1: BGV ciphertext calculation

Satisfy full homomorphisms. Input the ciphertext calculation key evk_1, the circuit C and the ciphertext c_1, and the output is the ciphertext calculation result c_1'.

2: Paillier ciphertext calculation

It satisfies add homomorphism, that is, the multiplication of ciphertext is equal to the addition of plaintext: $D(E(m1) \cdot E(m2)) = m1 + m2$. Since it supports additive homomorphism, Paillier algorithm can also support multiplication homomorphism, that is, multiplication of ciphertext and plaintext. The ciphertext calculation is as follows:

$$\begin{cases} c1 \equiv g^{m1} \cdot r_1^n \bmod n^2 \\ c2 \equiv g^{m2} \cdot r_2^n \bmod n^2 \end{cases}$$
$$\Rightarrow c1 \cdot c2 \equiv g^{m1} \cdot g^{m2} \cdot {r_1}^n \cdot {r_2}^n \bmod n^2$$
$$\Rightarrow c1 \cdot c2 \equiv g^{m1+m2} \cdot (r_1 \cdot r_2)^n \bmod n^2$$

$c_1 \cdot c_2$ can be regarded as $m = m1 + m2$ encrypted ciphertext, and the decryption result of $c_1 \cdot c_2$ is m.

Step 5: Decryption steps. Use adaptive ciphertext data selection to filter the ciphertext and perform homomorphic decryption.

1: BGV decryption

Input private key $s = sk_1$, ciphertext c_1, output decrypted plaintext $m_1' = \left[\frac{t}{q}[c_1[0] + c_1[1] \cdot s]_q\right]_t$.

2: Paillier decryption

Input private key $s = sk_2$, ciphertext c_1, output decrypted plaintext $m_2 = L\left({c_2}^\lambda \bmod n^2\right)\mu \bmod n$.

Step 6: Speech reconstruction. The decrypted speech matrix is restored to a complete decrypted speech.

4 Encryption Performance Analysis

In order to evaluate the performance and efficiency of the proposed speech encryption scheme, some unequal-length speeches in the Chinese speech database THCHS-30 [22] opened by Tsinghua University were selected as the test speech for this experiment for encryption and decryption. Among them, S1.wav is a 8s speech, S2.wav is a 6s speech, and S3.wav is a 4s speech. Use PyCharm platform tools to perform speech preprocessing to obtain adaptively classified speech data as a database to realize adaptive homomorphic encryption of speech data.

Experimental hardware environment: Intel(R) Core (TM) i5-8250U CPU, 1.80GHz, RAM 12GB.

Software Environment: Windows 10, PyCharm, Matlab R2017b.

4.1 Correlation Analysis

Correlation analysis [15], as a data statistical method, is widely used in the performance evaluation of speech encryption algorithms. If the value of the correlation coefficient is around +1 or −1, it indicates that the two speech signals are highly correlated; if the value of the correlation coefficient of the two speech signals is around 0, it means that the correlation between the two speech signals is extremely poor. The correlation coefficient expression is shown in Equation (5), and its correlation expression is as follows:

$$r_{xk} = \frac{C(x,k)}{\sqrt{V(x)}\sqrt{V(k)}} \tag{5}$$

$$C(x,k) = \frac{\sum (x-\bar{x})(k-\bar{k})}{N-1} \tag{6}$$

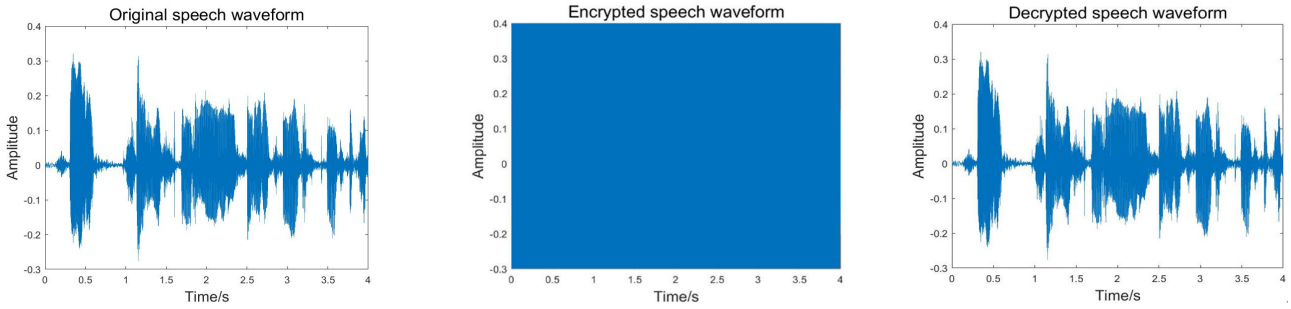
$$V(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2 \tag{7}$$

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{8}$$

where $\bar{x}$, $\bar{k}$ are the mean values of the original speech signal x and the encrypted speech signal k respectively; $C(x,k)$ is the covariance between the original speech signal x and the encrypted speech signal k; $V(x)$ and $V(k)$ represent the variance between the original speech x and the encrypted speech k.

Figure 5 shows the waveform of original speech, encrypted speech and decrypted speech of the S3.wav as an example. Figure 5(a), Figure 5(b) and Figure 5(c) are waveform diagrams of original speech, encrypted speech and decrypted speech respectively.

It can be seen from Figure 5(b) that no speech waveform features can be seen in the encrypted speech, so the encryption effect is good. In order to further reflect the anti-statistical analysis attack performance of the encryption algorithm, the correlation coefficient before and after the speech encryption is analyzed, and the Pearson correlation coefficient is calculated by Equation (5) to measure the correlation between the signals. Table 2 shows



(a) Original speech (b) Encrypted speech (c) Decrypted speech

Figure 5: Waveform diagram of original speech, encrypted speech and decrypted speech

the correlation analysis of the original speech, encrypted speech and decrypted speech of different durations. Figure 6 shows the correlation comparison before and after speech encryption.

Table 2: Correlation analysis of speech

File	Original & Encrypted	Original & Decrypted
S1.wav	-0.0043	0.9890
S2.wav	-0.0032	0.9920
S3.wav	0.0018	0.9983
Average	0.0031	0.9931

If the correlation coefficient is around $+1$ or -1, it indicates that the two speech signals are highly correlated. If the correlation coefficient between two speech signals is around 0, it means that the correlation between the two speech signals is extremely poor. It can be seen from Table 2 and Figure 6 that the correlation coefficient between the original speech and the encrypted speech is close to 0, indicating that the original speech and the encrypted speech are unrelated and the speech encryption performance is good. The correlation coefficient between the original speech and the decrypted speech is between -1 and $+1$, indicating that the recovery and reconstruction performance of the speech is extremely strong, and basically can achieve lossless recovery.

4.2 SNR and SNRseg

Signal-to-noise ratio (SNR) [16], as one of the most common and direct methods to verify the performance of data encryption algorithms. It is mainly used to measure the noise content and distortion degree of signals in encrypted data, and is widely used in multimedia data encryption. The expression of SNR is shown in Equation (9).

$$SNR=10\times\log_{10}\frac{\sum_{i=0}^{L}x_i^2}{\sum_{i=1}^{L}[x_i-y_i]^2} \tag{9}$$

where L represents the number of samples; x_i stands for the original speech signal; y_i stands for encrypted speech signal. The higher the SNR is, the less noise is generated. Generally speaking, the smaller the SNR is, the greater the noise in the encrypted signal is, the higher the distortion degree is and the better the encryption quality is.

Segmented SNR (SNRseg) [16] is the average of short-frame SNR. This is one of the widely used objective evaluation measurement methods, which can be used to estimate the quality of the speech signal. The lower the SNRseg value, the higher the encryption noise and the better the encryption effect. The expression of the SNRseg function is shown in Equation (10):

$$SNR_{seg}=\frac{10}{M}\times\sum_{m=0}^{M-1}\log_{10}\frac{\sum_{n=Lm}^{Lm+L-1}x_i^2}{\sum_{n=Lm}^{Lm+L-1}[x_i-y_i]} \tag{10}$$

where M represents the number of frames in the speech signal.

Table 3: SNR and SNRseg of encrypted speech

File	SNR (dB)	SNRseg (dB)
S1.wav	-40.1540	-41.1022
S2.wav	-38.4883	-40.5567
S3.wav	-51.0261	-53.6930
Average	-44.8495	-45.1173

The proposed scheme performs SNR and SNRseg tests on encrypted speech data of different durations. It can be seen from Table 3 that the SNR and SNRseg values of the encrypted speech obtained from the experimental results are lower, indicating that the proposed encryption scheme has higher encryption quality and stronger security.

4.3 Security Analysis

In general, the security analysis must be satisfied when a new encryption scheme is proposed. A perfect encryption scheme should be robust against all kinds of cryptanalysis attacks (i.e., statistical attack, entropy attack, chosen-plaintext attack, etc). Therefore, this paper conducts some security analyses to demonstrate the effectiveness of the proposed algorithm.

1) Statistical Attack

If the encryption performance of a speech encryption system is well, the encrypted speech statistics

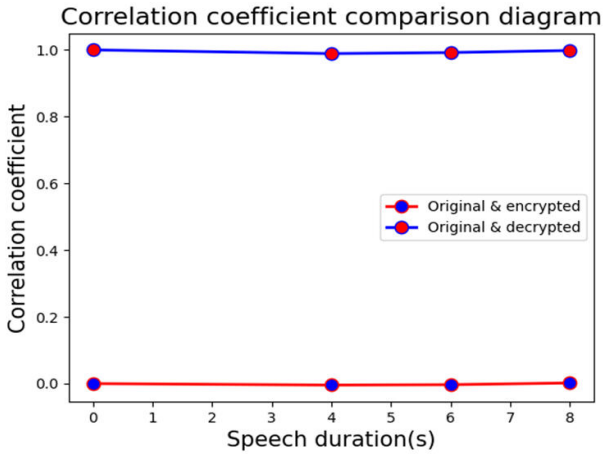


Figure 6: Correlation comparison before and after speech encryption

histogram [18] should be evenly distributed. This section takes S3.wav as an example for encryption performance analysis. Figure 7 shows the amplitude histogram of the original speech and the encrypted speech.

It can be seen from Figure 7 that the amplitude histogram of the original speech in Figure 7(a) has irregular statistical features. The amplitude histogram distribution of encrypted speech in Figure 7(b) is relatively stable without large ups and downs, which has a good masking effect on the statistical features of the speech data. It can be seen from Figure 7(b) that the encrypted speech data has poor correlation and little statistical information, indicating that the proposed encryption scheme is sufficient to resist statistical attack.

2) Entropy Attack

Information entropy analysis [10] is mainly used for the error rate of encrypted speech data. Generally, the value of information entropy is proportional to the error rate of speech. The higher the information entropy of encrypted speech data, the better the effect of speech encryption. The calculation expression of information entropy is shown in Equation (11):

$$H = -\sum_{K=0}^{S} p(k)\log_2 p(k) \tag{11}$$

where $p(k)$ is the input speech data, S represents the number of sampling points.

For each speech file, if the entropy value of the encrypted speech data is close to 16, it indicates that the speech encryption system has better encryption effect and higher security. Table 4 shows the calculation of information entropy for speech data of different durations.

It can be seen from Table 4 that the information entropy of encrypted speech data is basically close to the expected value of 16, indicating that the proposed encryption scheme has high security and is sufficient to resist entropy attack.

Table 4: Information entropy analysis of speech

File	Original speech	Encrypted speech
S1.wav	11.6241	15.4649
S2.wav	11.7700	15.6166
S3.wav	12.2390	15.7770
Average	11.5956	15.6594

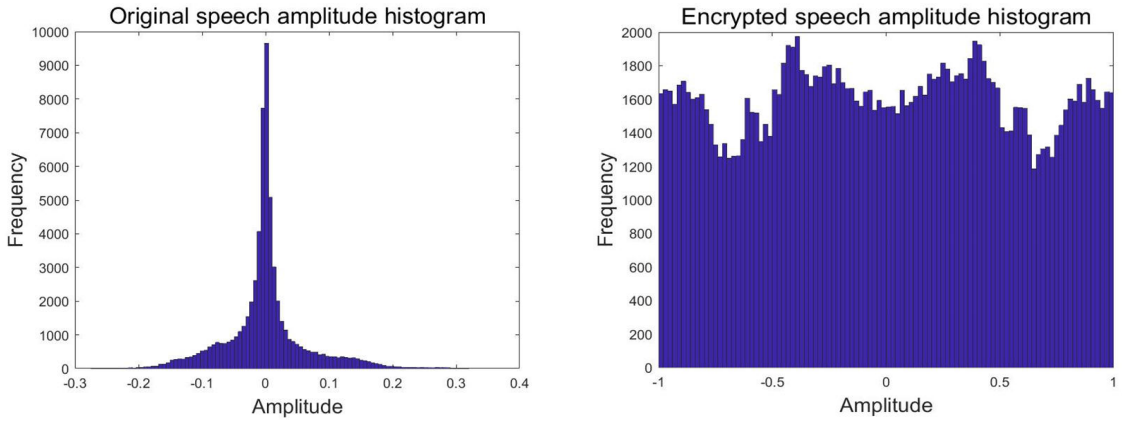
3) Chosen-plaintext Attack

The number of samples change rate (NSCR) [16] is an evaluation index of chosen-plaintext attack, which is widely used in the field of speech encryption. It reflects the proportion of the data points in the same position of two speech data to the whole data point. NSCR reflects the proportion of data points that are not equal in the same position of two speech data that are not equal to the entire data point. If NSCR is approximately equal to 100%, it is considered that the encryption algorithm has high performance and can resist various plaintext attacks. Table 5 shows the sample rate of change for different durations of speech.

Table 5: NSCR of speech

File	NSCR (%)
S1.wav	100
S2.wav	99.999
S3.wav	99.996
Average	99.998

It can be seen from Table 5 that the NSCR values obtained by the proposed scheme are all close to 100%, indicating that the encrypted speech data



(a) Original speech (b) Encrypted speech

Figure 7: Amplitude histogram of original speech and encrypted speech

sample points are diametrically opposite to the original speech, and the proposed scheme can effectively resist differential attacks.

5 Experimental Analysis

5.1 Adaptive Classification

The adaptive classifier proposed in this paper has obtained 3 decision points after 3 iterations. The decision point 1 that is less than (equal to) -1.0124 is divided into $+1$ category, the rest are divided into -1 category, and the weight of classifier Y_1 is 0.5. The decision point 2 is greater than (equal to) 1.0124 and divided into $+1$ category, the rest are divided into -1 category, and the weight of classifier Y_2 is 0.3. The decision point 3 is less than (equal to) 2.4492 is divided into $+1$ category, the rest are divided into -1 category, the weight of classifier Y_3 is 0.4.

This study loads the speech data as a sequence object, i.e. one-dimensional arrays, each with a time label. Confirm that the datasets have been loaded correctly with the summary data in Figure 8 and visualize these data as the dataset waveforms as shown in Figure 9.

By observing the density of the training data set, we can further understand the residual error of the data structure analysis model. Ideally, the distribution of residuals should follow a Gaussian distribution with zero mean. The residual is calculated by subtracting the predicted value from the actual value. Figure 10 shows the energy distribution of the sample data set used in this article, and Figure 11 shows the residual density of the designed classifier training model.

It can be seen from Figure 11 that the residual error of the adaptive classifier designed in this paper is basically Gaussian, the model has a small deviation, and the mean value basically tends to zero. If there is any autocorrelation in the residuals, it means there is an opportunity to improve the model. Ideally, if the model fits properly, no autocorrelation should be retained in the residuals. When training the classifier, first extract the first 50 data of the sample for autocorrelation analysis, and the autocorrelation coefficient is shown in Figure 12; when all the data of a sample is input to train the classifier, the autocorrelation coefficient is shown in Figure 13.

It can be seen from Figure 12 and Figure 13 that the autocorrelation coefficient of training a sample of all data is much lower than that of training a sample of a small amount of data. The more samples in the training set, the more the autocorrelation tends to zero. Figure 13 shows that all autocorrelations have been captured in the training model, and there is no autocorrelation in the residuals. Therefore, the training model has passed all the standards, and this model can be saved as an adaptive classifier for subsequent use.

5.2 Speech Encryption and Decryption Efficiency Analysis

The complexity of the speech encryption system and the efficiency of speech encryption and decryption are mutually restricted. Existing algorithms often ignore the speech encryption and decryption time when ensuring key security, and are not suitable for massive speech encrypted data. Table 6 shows the time efficiency analysis of encryption and decryption of speech data with different durations.

Table 6: Efficiency of speech encryption and decryption

File	Encryption time(s)	Decryption time(s)
S1.wav	15.7421	8.8643
S2.wav	11.3554	5.9435
S3.wav	8.1106	4.1652
Average	1.9560	1.0541

It can be seen from Table 6 that the encryption algorithm use about 1.9560 s to encrypt speech per second, and the decryption algorithm takes about 1.0541 s to encrypt speech per second, indicating that the proposed

```
count     64000.000000
mean       9036.312578
std        2038.179647
min           0.000000
25%        8356.000000
50%        9013.000000
75%        9525.000000
max       19552.000000
Name: 0, dtype: float64
```

Figure 8: Summary data

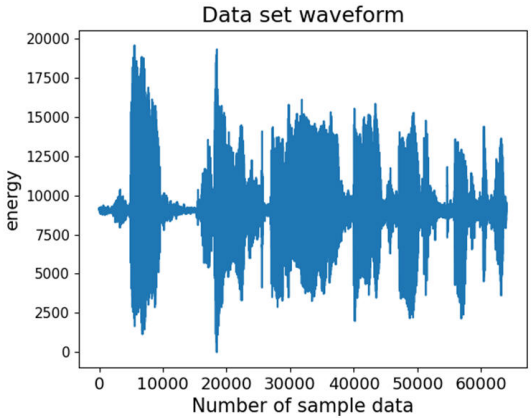


Figure 9: Dataset waveform

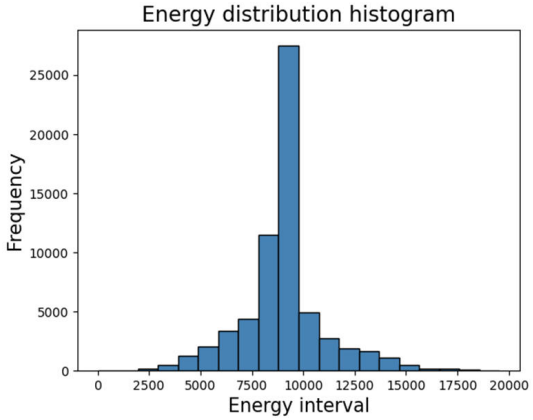


Figure 10: Histogram of energy distribution

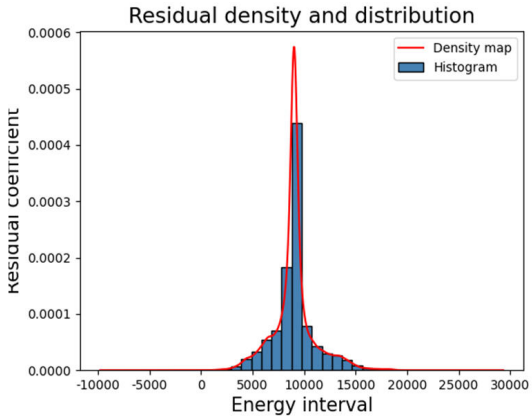


Figure 11: Residual error density and distribution

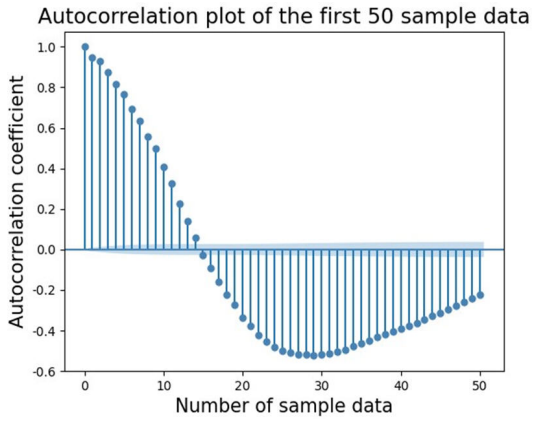


Figure 12: Autocorrelation plot of the first 50 sample data

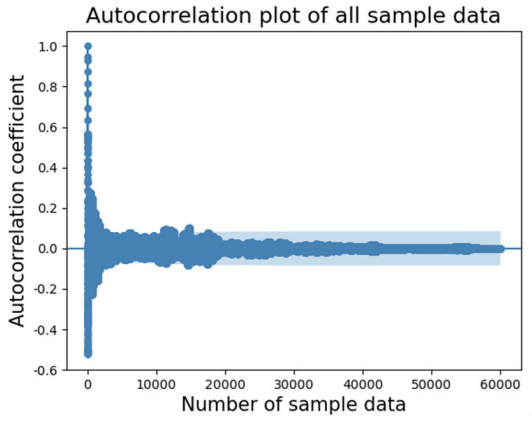


Figure 13: Autocorrelation plot of all sample data

scheme has good encryption and decryption efficiency.

5.3 Comparative Analysis with Existing Encryption Schemes

This section compares the experimental results with the improved probabilistic statistics addition homomorphic algorithm, El-Gamal, probabilistic homomorphic speech encryption algorithm in the existing scheme [6,15,16] and BFV speech homomorphic encryption scheme to objectively and accurately evaluates the proposed scheme. The experimental comparisons all adopt the speech data with a duration of 4s, and take the average value of each item for comprehensive evaluation as shown in Table 7.

It can be seen from Table 7 that compared with Ref. [6, 15, 16] and BFV homomorphic encryption sys-

Table 7: Performance comparison

Evaluation index	Proposed	Ref. [15]	Ref. [6]	Ref. [16]	Ref. [1]	Ref. [14]	BFV
Key size	2×256	-	-	196	-	$L \times 16$	128
key space	$2^{2\times256}$	-	-	4×2^{196}	-	$2^{L\times16}$	2^{128}
SNR (dB)	-44.8495	-29.96	-35.0224	-45.0206	-	-47.4953	-
SNRseg (dB)	-45.1173	-	-38.0201	-45.2222	-	-	-
Correlation coefficient original & encrypted	0.9931	0.9438	-	0.7386	0.9901	0.9981	-
Correlation coefficient original & decrypted	0.0031	0.0027	-	0.0115	0.0008	0.0032	-
Encryption time (s)	8.1106	25.3075	-	5.7208	2.3005	-	13.0878
Decryption time (s)	4.1652	24.2481	-	29.0230	2.8775	-	4.1264
Cipher expand	24.7519	6.6667×10^6	-	-	-	-	26.5397
Statistical attack	✓	✓	×	✓	✓	✓	✓
Entropy attacks	✓	×	×	×	×	×	×
Chosen-plaintext attack	✓	✓	×	×	✓	✓	✓

tem, the proposed scheme is generally superior than other speech homomorphic encryption algorithms. Compared with Ref. [14], the adaptive speech homomorphic encryption scheme proposed in this paper has a larger key space than adaptive speech encryption model. This is mainly because the proposed scheme performs adaptive classification on the speech data at first, and then the classified data of different types are separately encrypted with different homomorphic encryption in parallel. While ensuring good encryption performance, the computational complexity and the ciphertext expansion are reduced. Compared with Ref. [1], the encryption and decryption efficiency is lower than that of the speech chaotic encryption algorithm, but the proposed scheme has higher security, and can realize the subsequent ciphertext operation to support the ciphertext speech retrieval system.

6 Conclusions

In this paper, an adaptive speech homomorphic encryption scheme based on energy in cloud storage is proposed, which solves the risks of data privacy exposure of traditional speech encryption methods, the low efficiency of existing speech homomorphic encryption schemes, and the poor adaptability of speech encryption schemes. The proposed scheme combines adaptive technology and homomorphic encryption technology to perform energy-based adaptive data classification and batch encryption of speech data, reducing the amount of the data encrypted by FHE to improve the efficiency of speech homomorphic encryption. Using the ciphertext calculation function supported by homomorphic encryption to realize speech data calculation operations in the ciphertext domain can greatly improve the security and calculation efficiency of speech retrieval and speech recognition systems. The analysis of the encryption/decryption capabilities of different test speech signals through multiple performance indicators such as correlation coefficient, SNR, and SNRseg shows that the proposed scheme can provide encrypted speech signals with low residual intelligibility. By comparing the performance with the existing scheme, the proposed scheme effectively improves the efficiency of speech homomorphic encryption, and can effectively improve the adaptability of the encryption scheme under the premise of keeping the algorithm complexity unchanged. It has higher security and lower ciphertext expansion, and can resist statistical attack, entropy attack, and chosen-plaintext attack.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61862041, 61363078). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] O. M. Al-Hazaimeh, "A new speech encryption algorithm based on dual shuffling henon chaotic map," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 3, pp. 2203–2210, 2021.

[2] H. Chen, W. Dai, M. Kim, "Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference," in *In the 2019 ACM SIGSAC Conference*, pp. 395–412, London, UK, November 2019.

[3] X. Feng, Y. Zhou, "English audio language retrieval based on adaptive speech-adjusting algorithm," *Complexity*, vol. 2021, pp. 2762180(1)–2762180(12), 2021.

[4] M. S. Hwang, E. F. Cahyadi, S. F. Chiou, C. Y. Yang, "Reviews and analyses the privacy-protection system for multi-server," *Journal of Physics: Conference Se-*

ries, vol. 1237, no. 2, p. 022091, IOP Publishing, 2019.

[5] I. Iliashenko, V. Zucca, "Faster homomorphic comparison operations for bgv and bfv," *Proceedings on Privacy Enhancing Technologies*, vol. 2021, no. 3, pp. 246–264, 2021.

[6] O. A. Imran, S. F. Yousif, I. S. Hameed, "Implementation of el-gamal algorithm for speech signals encryption and decryption," *Procedia Computer Science*, vol. 167, no. 3, pp. 1028–1037, 2020.

[7] H. Jahanshahi, A. Yousefpour, J. M. Munoz-Pacheco, "A new fractional-order hyperchaotic memristor oscillator: Dynamic analysis, robust adaptive synchronization, and its application to voice encryption," *Applied Mathematics and Computation*, vol. 383, no. 2, pp. 125310, 2020.

[8] H. Leng, H. Chen, "Adaptive hdg methods for the brinkman equations with application to optimal control," *Journal of Scientific Computing*, vol. 87, no. 46, pp. 2–34, 2021.

[9] Z. Y. Li, X. L. Gui, Y. J. Gu, X. S. Li, H. J. Dai, X. J. Zhang, "Survey on homomorphic encryption algorithm and its application in the privacy-preserving for cloud computing (in chinese)," *Ruan Jian Xue Bao/Journal of Software* (in Chinese), vol. 29, no. 7, pp. 1830–1851, 2018.

[10] L. Liu, Z. Cao, C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[11] S. Najam, M. U. Rehman, J. Ahmed, "Data encryption scheme based on adaptive system," in *Global Conference on Wireless and Optical Technologies (GCWOT)*, pp. 1–4, University of Malaga, Spain, October 2020.

[12] C. Orlandi., P. Scholl, S. Yakoubov, "The rise of paillier: Homomorphic secret sharing and public-key silent ot," in *Advances in Cryptology – EUROCRYPT 2021*, pp. 678–708, Zagreb, Croatia, October 2021.

[13] A. Siswanto, C. Y. Chang, S. M. Kuo, "Multirate audio-integrated feedback active noise control systems using decimated-band adaptive filters for reducing narrowband noises," *Sensors*, vol. 20, no. 22, pp. 6693–6705, 2020.

[14] H. I. Shahadi. "Covert communication model for speech signals based on an indirect and adaptive encryption technique," *Computers and Electrical Engineering*, vol. 68, no. 4, p. 425–436, 2018.

[15] C. Shi, H. Wang, Y. Hu, "A speech homomorphic encryption scheme with less data expansion in cloud computing," *KSII Transactions on Internet and Information Systems*, vol. 13, no. 5, pp. 2588–2609, 2019.

[16] C. Shi, H. Wang, Q. Qian, H. Wang, "Privacy protection of digital speech based on homomorphic encryption," in *Cloud Computing and Security (ICCCS)*, pp. 365–376, Nanjing, China, July 2016.

[17] Y. Tang, B. Zhu, X. Ma, "Decoding homomorphically encrypted flac audio without decryption," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 675–679, Brighton, UK, May 2019.

[18] L. Teng, H. Li, J. Liu, "An efficient and secure ciphertext retrieval scheme based on mixed homomorphic encryption and multi-attribute sorting method under cloud environment," *International Journal of Network Security*, vol. 20, no. 5, pp. 872–878, 2018.

[19] L. Teng, H. Li, S. Yin, "IM-Mobishare: An improved privacy preserving scheme based on asymmetric encryption and bloom filter for users location sharing in social network," *Journal of Computers (Taiwan)*, vol. 30, no. 3, pp. 59–71, 2019.

[20] P. Thaine, G. Penn, "Extracting mel-frequency and bark-frequency cepstral coefficients from encrypted signals," in *INTERSPEECH 2019*, pp. 3715–3719, Graz, Austria, September 2019.

[21] A. V. Tutueva, E. G. Nepomuceno, A. I. Karimov, "Adaptive chaotic maps and their application to pseudo-random numbers generation," *Chaos, Solitons & Fractals*, vol. 133, no. 3, p. 109615, 2020.

[22] D. Wang, X. Zhang, "Thchs-30: A free chinese speech corpus," *arXiv preprint arXiv:1512.01882*, 2015.

[23] Q. Wu, M. Wu, "Adaptive and blind audio watermarking algorithm based on chaotic encryption in hybrid domain," *Symmetry*, vol. 10, no. 7, p. 284, 2018.

[24] A. Ww, C. Dsb, "The improved adaboost algorithms for imbalanced data classification," *Information Sciences*, vol. 563, no. 7, pp. 358–374, 2021.

[25] S. Yin, J. Liu, L. Teng, "Improved elliptic curve cryptography with homomorphic encryption for medical image encryption," *International Journal of Network Security*, vol. 22, no. 3, pp. 419–424, 2020.

[26] X. Zhu, T. K. Han, M. Kim, "Privacy-preserving multimedia data analysis," *The Computer Journal*, vol. 64, no. 7, pp. 991-992, 2021.

Biography

Qiu-yu Zhang Researcher/Ph.D. supervisor, graduated from Gansu University of Technology in 1986, and then worked at school of computer and communication in Lanzhou University of Technology. He is vice dean of Gansu manufacturing information engineering research center, a CCF senior member, a member of IEEE and ACM. His research interests include network and information security, information hiding and steganalysis, multimedia communication technology.

Yu-jiao Ba is currently a master student of the School of Computer and Communication, Lanzhou University of Technology, China. She received the BS degrees in computer science and technology from Lanzhou University of Technology, Gansu, China, in 2019. Her research

interests include network and information security, audio signal processing and application, multimedia data security.

Quantum Synchronizable Codes From Sextic Cyclotomy

Tao Wang, Xueting Wang, Qian Liu, and Tongjiang Yan
(Corresponding author: Tongjiang Yan)

College of Science, China University of Petroleum
Qingdao 266555, Shangdong, China.
Email: yantoji@163.com

(Received Oct. 22, 2021; Revised and Accepted Apr. 15, 2022; First Online May 1, 2022)

Abstract

Quantum synchronizable codes can be used to correct the effects of both quantum noise on qubits and misalignments in block synchronization. This paper contributes to constructing quantum synchronizable codes from the dual-containing cyclic codes obtained by sextic cyclotomy. We show that these quantum synchronizable codes possess good synchronization capabilities, which can always attain the upper bound, and good error-correcting capability towards bit errors and phase errors when the corresponding cyclic codes are optimal or almost optimal.

Keywords: Cyclic Codes; Quantum Synchronizable Codes; Sextic Cyclotomy

1 Introduction

In decades, quantum information theory has made great progress in quantum information and quantum communication, especially in quantum error-correcting codes [16]. However, the studies on quantum error-correcting codes tend to just focus on the simplest Pauli errors on qubits, which roughly corresponds to additive noise in classical encoding theory [3,9,17]. Meanwhile, the misalignment in block synchronization can also cause catastrophic failure in quantum information transmission. This kind of error occurs due to the fact that the information processing devices misidentify the boundaries of an information qubit stream. For instance, suppose that the quantum information can be expressed by an ordered sequence of information block and each chunk of information is encoded into a block of consecutive three qubits in a stream of qubit $|q_i\rangle$, $i \in I$, where I is an indexed set. If three blocks of information are encoded, we have 9 ordered qubits $(|q_0\rangle|q_1\rangle|q_2\rangle|q_3\rangle|q_4\rangle|q_5\rangle|q_6\rangle|q_7\rangle|q_8\rangle)$, then each of the three blocks $(|q_0\rangle|q_1\rangle|q_2\rangle)$, $(|q_3\rangle|q_4\rangle|q_5\rangle)$ and $(|q_6\rangle|q_7\rangle|q_8\rangle)$ forms an information chunk. Suppose the synchronization system was established at the beginning of information transmission, but synchronization may be lost during the quantum communications or quantum computations. The misalignment occurs when the receiver incorrectly locates the boundary of each block of data by a certain number of positions towards the left or right. For example, the receiver wrongly read out $(|q_5\rangle|q_6\rangle|q_7\rangle)$ instead of the correct information chunk $(|q_6\rangle|q_7\rangle|q_8\rangle)$. For more details, see [4].

As a subclass of quantum error-correcting codes, quantum synchronizable codes (QSCs) can be used to prevent both the interference of quantum noises on qubits and misalignments in block synchronization. In order to ensure information security, it is of great significance to study the construction of QSCs. In 2013, Fujiwara *et al.* [5, 6] proposed the framework of quantum block synchronization and gave the first example of QSCs. In 2014, Xie *et al.* [18] used quadratic residue codes to produce binary QSCs which attain the upper bound on synchronization capabilities. Recently, Li and Yue [13] obtained two families of QSCs with good error-correcting performance. Some further studies about QSCs can be seen in [12, 14].

Although we now have the theoretical framework of QSCs, there exists only a few families of QSCs in the literature. Cyclotomic classes can be used to constructed self-dual codes [7, 11], which have important appliance in constructing QSCs. Hence, we use the cyclic codes obtained by sextic cyclotomy to construct QSCs in this work. This paper is arranged as follows. In Section 2, we review some general conclusions of cyclic codes and cyclotomic classes. In Section 3, we construct some dual-containing cyclic codes and discuss their minimum Hamming distance. In Section 4, we construct two classes of QSCs and discuss their synchronization capabilities and error-correcting performance. Finally, some concluding remarks are given in Section 5.

2 Preliminaries

2.1 Cyclic Codes and Dual Codes

Let $\mathbb{F}_q$ be a finite field with q elements, where q is a prime power. An $[n, k, d]_q$ linear code C is a k-dimensional subspace of the n-dimensional vector space over $\mathbb{F}_q$ such that

$\min\{\mathrm{wt}(v)|v \in C,\ v \neq 0\} = d$, where $\mathrm{wt}(v)$ is the Hamming weight of v. A linear code with parameters $[n,k,d]_q$ is called optimal if and only if its minimum Hamming distance reaches the Hamming bound, see e.g. [2]. A linear code with parameters $[n,k,d]_q$ is called almost optimal means that the code with parameters $[n,k,d+1]_q$ is optimal. An $[n,k]_q$ cyclic code C is a linear code with the property that if a codeword $c = (c_0, c_1, \cdots, c_{n-1}) \in C$, then $(c_{n-1}, c_0, \cdots, c_{n-2}) \in C$. It is known that C can be seen as an principal ideal $\langle g(x)\rangle$ in $\mathbb{F}_q[x]/(x^n-1)$. The polynomial $g(x)$ with degree $n-k$ is a monic divisor of x^n-1, and it is called the generator polynomial of C. The polynomial $h(x) = x^n - 1/g(x)$ is called the parity-check polynomial of C.

The Euclidean inner product between two codewords $x = (x_0, x_1, \ldots, x_{n-1})$ and $y = (y_0, y_1, \ldots, y_{n-1})$ is defined by $(x,y) = \sum_{i=0}^{n-1} x_i y_i$. The Euclidean dual code $C^\perp = \{x \in \mathbb{F}_q^n | (x,c) = 0, \forall c \in C\}$ of C is also a cyclic code [10], and the generator polynomial of $C^\perp$ has the form

$$\tilde{h}(x) = h(0)^{-1} x^k h\left(x^{-1}\right), \tag{1}$$

which is called the reciprocal polynomial of $h(x)$.

Let $C_1 = \langle g_1(x)\rangle$ and $C_2 = \langle g_2(x)\rangle$ be two cyclic codes with parameters $[n,k_1]_q$ and $[n,k_2]_q$ respectively. If $C_1 \subseteq C_2$, then C_2 is said to be C_1-containing, and C_2 is called the augmented code of C_1. If $C_2^\perp \subset C_2$, C_2 is called dual-containing.

2.2 Sextic Cyclotomic Classes

Let $n = 12m+7$ be an odd prime and γ be a fixed primitive element in $\mathbb{F}_n$. Then the sextic cyclotomic classes $C_0^{(6,n)}$, $C_1^{(6,n)}$, $\ldots, C_5^{(6,n)}$ in $\mathbb{F}_n$ are

$$C_i^{(6,n)} = \{\gamma^{6j+i} | 0 \le j \le \frac{n-1}{6} - 1\}, \text{ for } i = 0, 1, \cdots, 5.$$

Trivially $C_i^{(6,n)} = \gamma^i C_0^{(6,n)}$ and $\mathbb{F}_n^* = \bigcup_{i=0}^5 C_i^{(6,n)}$, where $\mathbb{F}_n^* = \mathbb{F}_n \backslash \{0\}$.

Lemma 1. *Let the notations be defined as above. Then*

$$C_i^{(6,n)} = -C_{i+3}^{(6,n)},$$

where $i = 0, 1, \cdots, 5$, and $i+3$ means $i+3 \pmod 6$.

Proof. Since γ is a fixed primitive element in $\mathbb{F}_n$, $-1 = \gamma^{\frac{n-1}{2}} = \gamma^{6m+3} \in C_3^{(6,n)}$. Then the result can be obtained immediately. □

From now on, we let $q \in C_0^{(6,n)}$, and η be a n-th primitive root of unity in $\mathbb{F}_{q^{\mathrm{ord}_n(q)}}$, where $\mathrm{ord}_n(q)$ is the multiplicative order of q modulo n. Let

$$g_i^{(6,n)}(x) = \prod_{j \in C_i^{(6,n)}} (x - \eta^j), \tag{2}$$

where $i = 0, 1, \cdots, 5$. It is known that $g_i^{(6,n)}(x) \in \mathbb{F}_q[x]$, and the factorization of $x^n - 1$ is

$$x^n - 1 = (x-1) \prod_{i=0}^{5} g_i^{(6,n)}(x).$$

3 Cyclic Codes from Sextic Cyclotomy

3.1 Dual-containing Codes

Let C_i and $\bar{C}_i$ be the cyclic codes over $\mathbb{F}_q$ generated by $g_i^{(6,n)}(x)$ and $\frac{x^n-1}{g_{i+3}^{(6,n)}(x)}$ respectively, $i = 0, 1, \cdots, 5$.

Lemma 2. *Let $n = 12m+7$ be an odd prime, where m is a nonnegative integer. Then*

$$(a)\ C_i^\perp = \bar{C}_i, \qquad (b)\ C_i^\perp \subset C_i. \tag{3}$$

Proof. By Equation (1), the reciprocal polynomial of $g_i^{(6,n)}(x)$ is

$$\tilde{g}_i^{(6,n)}(x) = \left(g_i^{(6,n)}(0)\right)^{-1} x^{\deg\left(g_i^{(6,n)}(x)\right)} g_i^{(6,n)}(x^{-1}).$$

Assume that

$$g_i^{(6,n)}(x) = (x - \eta^{e_{i_1}})(x - \eta^{e_{i_2}}) \ldots (x - \eta^{e_{i_{2m+1}}}),$$

where e_{i_j} runs over $C_i^{(6,n)}$ and each element appears only once. Then

$$\begin{aligned}\tilde{g}_i^{(6,n)}(x) =& (-\eta^{e_{i_1}})^{-1}(-\eta^{e_{i_2}})^{-1} \ldots (-\eta^{e_{i_{2m+1}}})^{-1} x^{2m+1} \\ & (x^{-1} - \eta^{e_{i_1}})(x^{-1} - \eta^{e_{i_2}}) \ldots (x^{-1} - \eta^{e_{i_{2m+1}}}) \\ =& (x - \eta^{-e_{i_1}})(x - \eta^{-e_{i_2}}) \ldots (x - \eta^{-e_{i_{2m+1}}}).\end{aligned}$$

According to Lemma 1 and Equation (2), we have

$$\tilde{g}_i^{(6,n)}(x) = g_{i+3}^{(6,n)}(x), \tag{4}$$

where $i = 0, 1, \cdots, 5$. Note that the parity-check polynomial of C_i is

$$h(x) = \frac{x^n - 1}{g_i^{(6,n)}(x)} = (x-1) \prod_{j \in F_n^* \backslash C_i^{(6,n)}} (x - \eta^j).$$

And by Equation (4),

$$\tilde{h}(x) = (x-1) g_i^{(6,n)}(x) g_{i+1}^{(6,n)}(x) g_{i+2}^{(6,n)}(x) g_{i+4}^{(6,n)}(x) g_{i+5}^{(6,n)}(x)$$

is the generator polynomial of $C_i^\perp$. This means $C_i^\perp = \bar{C}_i$. Moreover, since $g_i^{(6,n)}(x) | \tilde{h}(x)$, we get $C_i^\perp \subset C_i$. □

In addition, let D_i and $\bar{D}_i$ be the cyclic codes over $\mathbb{F}_q$ generated by $g_i^{(6,n)}(x) g_{i+1}^{(6,n)}(x) g_{i+2}^{(6,n)}(x)$ and $(x-1) g_i^{(6,n)}(x) g_{i+1}^{(6,n)}(x) g_{i+2}^{(6,n)}(x)$ respectively, for any $i \in \{0,1,2,3,4,5\}$. By using the similar method in Lemma 2, we have the following Lemma.

Lemma 3. *Let $n = 12m + 7$ be an odd prime. Then*

$$(a)\ D_i^{\perp} = \bar{D}_i, \qquad (b)\ D_i^{\perp} \subset D_i. \tag{5}$$

Proof. The proof is straightforward from Lemma 2. □

Theorem 1. *Let d_i be the minimum Hamming distance of the cyclic code D_i. Then $d_i^2 - d_i + 1 \geq n$, where n is the length of D_i.*

Proof. Let $d(x)$ be a codeword in D_i with minimum Hamming weight d. From Equation (2),

$$d(x) = \prod_{i \in C_i^{(6,n)} \cup C_{i+1}^{(6,n)} \cup C_{i+2}^{(6,n)}} (x - \eta^i) s(x),$$

where $s(x) \in \mathbb{F}_q[x]$, $\deg(s(x)) < \frac{n+1}{2}$. Since η^i are the roots of $d(x) = 0$, $d(x^{-1}) = 0$ have roots η^{-i}, where $i \in C_i^{(6,n)} \cup C_{i+1}^{(6,n)} \cup C_{i+2}^{(6,n)}$. Then we have $-1 \in C_{i+3}^{(6,n)}$, we then have $d(x^{-1})$ is a codeword in D_{i+3} with minimum Hamming weight d. Therefore, $d(x)d(x^{-1})$ is a codeword in $D_i \cap D_{i+3}$, which means $d(x)d(x^{-1})$ a multiple of

$$g_i^{(6,n)}(x) g_{i+1}^{(6,n)}(x) g_{i+2}^{(6,n)}(x) g_{i+3}^{(6,n)}(x) g_{i+4}^{(6,n)}(x) g_{i+5}^{(6,n)}(x) = \frac{x^n - 1}{x - 1} = \sum_{j=0}^{n-1} x^j.$$

Then the weight of codewrod $d(x)d(x^{-1})$ is n. Since there are d terms equal to some nonzero elements of $\mathbb{F}_q$ in $d(x)$, we have $d^2 - d + 1 \geq n$. The desired result follows. □

Example 1. *Let $n = 12m + 7$ and $q \in C_0^{(6,n)}$. Table 1 gives some examples of cyclic codes and their duals, and some of them are optimal or almost optimal. All computations have been done by MAGMA [1]. Obviously, the minimum Hamming distance of D_i in Table 1 satisfy the bound in Theorem 1.*

Remark 1. *From Lemmas 2 and 3, we obtain two classes of dual-containing cyclic codes C_i and D_i. Furthermore, for any $i = \{0, 1, 2\}$, the cyclic codes with generator polynomial $g_i^{(6,n)}(x) g_{i+j}^{(6,n)}(x)$ are dual-containing codes, where $j \in \{0, 1, 2, 4, 5\}$ and $j \neq i$.*

3.2 Augmented Cyclic Codes

In order to obtain the augmented cyclic codes of C_i and D_i, we need the concept of cyclotomic cosets. The q-cyclotomy coset modulo n containing the integer s is defined as

$$C_{(s,n)} = \{sq^i \pmod{n} | i \in \mathbb{N}\}, \tag{6}$$

where $\mathbb{N}$ is the set of all nonnegative integers. It is notable that $s \in \{0, 1, 2, \ldots, n-1\}$. Then the unique irreducible minimal polynomial of η^s in $\mathbb{F}_q[x]$ is

$$M_s(x) = \prod_{i \in C_{(s,n)}} (x - \eta^i). \tag{7}$$

Lemma 4. *[15] Let n be an odd prime, and the size of $C_{(1,n)}$ be ℓ. Then the size of any cyclotomic coset $C_{(s,n)}$ is ℓ.*

Theorem 2. *Let $n = 12m + 7$ be an odd prime, $C_i = \langle g_i^{(6,n)}(x) \rangle$. If the size of $C_{(1,n)}$ is ℓ, the generator polynomial $g_i^{(6,n)}(x)$ can be expressed as*

$$g_i^{(6,n)}(x) = \prod_{j=1}^{t} M_{i_j}(x),$$

where $t = \frac{n-1}{6\ell}$.

Proof. By Lemma 4, the size of $C_{(s,n)}$ is ℓ. Let $q = \gamma^{6k} \in C_0^{(6,n)}$ be a prime power, where $k \in \{1, 2, \ldots, \frac{n-7}{6}\}$. As γ is the fixed primitive element in $\mathbb{F}_n$, it is easy to deduce that $C_{(s,n)} \subseteq C_i^{(6,n)}$. Since $\bigcup_{i=0}^{5} C_i^{(6,n)} = \bigcup_{s=1}^{n-1} C_{(s,n)} = \mathbb{F}_n^*$, we have $C_i^{(6,n)} = \bigcup_{j=1}^{t} C_{(i_j,n)}$, where $i_1, i_2, \cdots, i_t \in \{1, 2, \cdots, n-1\}$ are some appropriate integers. Furthermore, we have $t = \frac{|C_i^{(6,n)}|}{|C_{(s,n)}|} = \frac{n-1}{6\ell}$, where $|C_{(s,n)}|$ means the size of $C_{(s,n)}$. By Equations (7) and (2), we have $g_i^{(6,n)}(x) = \prod_{j=1}^{t} M_{i_j}(x)$. □

Example 2. *Consider the sextic cyclotomic classes $C_i^{(6,n)}$ in $\mathbb{F}_{127}$.*

$$\begin{aligned}
C_0^{(6,127)} &= \{1, 47, 50, 64, 87, 25, 32, 107, 76, 16, 117, 38, 8, \\
&\quad 122, 19, 4, 61, 73, 2, 94, 100\}, \\
C_1^{(6,127)} &= \{6, 28, 46, 3, 14, 23, 65, 7, 75, 96, 67, 101, 48, 97, \\
&\quad 114, 24, 112, 57, 12, 56, 92\}, \\
C_2^{(6,127)} &= \{36, 41, 22, 18, 84, 11, 9, 42, 69, 68, 21, 98, 34, 74, \\
&\quad 49, 17, 37, 88, 72, 82, 44\}, \\
C_3^{(6,127)} &= \{89, 119, 5, 108, 123, 66, 54, 125, 33, 27, 126, 80, \\
&\quad 77, 63, 40, 102, 95, 20, 51, 111, 10\}, \\
C_4^{(6,127)} &= \{26, 79, 30, 13, 103, 15, 70, 115, 71, 35, 121, 99, \\
&\quad 81, 124, 113, 104, 62, 120, 52, 31, 60\}, \\
C_5^{(6,127)} &= \{29, 93, 53, 78, 110, 90, 39, 55, 45, 83, 91, 86, 105, \\
&\quad 109, 43, 116, 118, 85, 58, 59, 106\}.
\end{aligned}$$

Let $\gamma = 3$ be the fixed primitive element of $\mathbb{F}_{127}$. Since $q = \gamma^{6K} \in C_0^{(6,127)}$, where $K \in \{0, 1, \cdots, 20\}$, the order of q modulo n is $\frac{|\gamma|}{\gcd(6K, |\gamma|)}$. If $K = 14$, then $q = 19$ and the order of q modulo n is 3. By Equation (6), we have

$$\begin{aligned}
&C_{(1,127)} = \{1, 19, 107\}, \quad C_{(2,127)} = \{2, 38, 87\}, \\
&C_{(4,127)} = \{4, 76, 47\}, \quad C_{(8,127)} = \{8, 25, 94\}, \\
&C_{(16,127)} = \{16, 50, 61\}, \quad C_{(32,127)} = \{32, 100, 122\}, \\
&C_{(64,127)} = \{64, 73, 117\}.
\end{aligned}$$

Thus $C_0^{(6,127)} = C_{(1,127)} \cup C_{(2,127)} \cup C_{(4,127)} \cup C_{(8,127)} \cup C_{(16,127)} \cup C_{(32,127)} \cup C_{(64,127)}$, which is equivalent to

$$g_0^{(6,127)}(x) = M_1(x) M_2(x) M_4(x) M_8(x) M_{16}(x) M_{32}(x) M_{64}(x).$$

Table 1: Dual-containing cyclic codes C_i and D_i

Codes	Dual codes	Comments
$C_i = [19, 16, 3]_7$	$C_i^{\perp} = [19, 3, 15]_7$	Both optimal [8]
$D_i = [19, 10, 7]_7$	$D_i^{\perp} = [19, 9, 8]_7$	Both almost optimal [8]
$C_i = [31, 26, 3]_2$	$C_i^{\perp} = [31, 5, 16]_2$	Both optimal [8]
$D_i = [31, 16, 7]_2$	$D_i^{\perp} = [31, 15, 8]_2$	D_i almost optimal, $D_i^{\perp}$ optimal [8]
$C_i = [43, 36, 5]_4$	$C_i^{\perp} = [43, 7, 27]_4$	Both optimal [8]
$D_i = [43, 22, 12]_4$	$D_i^{\perp} = [43, 21, 12]_4$	D_i almost optimal [8]
$C_i = [67, 56, 6]_9$	$C_i^{\perp} = [67, 11, 44]_9$	C_i almost optimal, $C_i^{\perp}$ optimal [8]
...	...	...

In this way, we have the following equations.

$$\begin{aligned}
C_1^{(6,127)} =& C_{(3,127)} \cup C_{(6,127)} \cup C_{(12,127)} \cup C_{(24,127)} \\
& \cup C_{(48,127)} \cup C_{(96,127)} \cup C_{(65,127)}, \\
C_2^{(6,127)} =& C_{(9,127)} \cup C_{(18,127)} \cup C_{(36,127)} \cup C_{(72,127)} \\
& \cup C_{(17,127)} \cup C_{(34,127)} \cup C_{(68,127)}, \\
C_3^{(6,127)} =& C_{(5,127)} \cup C_{(10,127)} \cup C_{(20,127)} \cup C_{(40,127)} \\
& \cup C_{(80,127)} \cup C_{(33,127)} \cup C_{(66,127)}, \\
C_4^{(6,127)} =& C_{(13,127)} \cup C_{(26,127)} \cup C_{(52,127)} \cup C_{(104,127)} \\
& \cup C_{(81,127)} \cup C_{(35,127)} \cup C_{(70,127)}, \\
C_5^{(6,127)} =& C_{(29,127)} \cup C_{(58,127)} \cup C_{(116,127)} \cup C_{(105,127)} \\
& \cup C_{(83,127)} \cup C_{(39,127)} \cup C_{(78,127)},
\end{aligned}$$

It is easy to deduce that the augmented cyclic code of C_i (or D_i) can be obtained by removing one or more irreducible factors of $g_i^{(6,n)}(x)$ (or $g_i^{(6,n)}(x)g_{i+1}^{(6,n)}(x)g_{i+2}^{(6,n)}(x)$). It is also notable that any augmented code obtained by this way is also dual-containing code. Furthermore, we have the following results.

Lemma 5. *Let $n = 12m+7$ be an odd prime, and C_i, D_i be the cyclic codes defined by above for $i \in \{0, 1, 2, 3, 4, 5\}$. If t in Theorem 2 is greater than 1, the following conclusions are established.*

1) *If $C = \langle \frac{g_i^{(6,n)}(x)}{\prod_{i\in A} M_i(x)} \rangle$, then $C_i \subset C$, where A is some nonempty subset of $\{i_1, i_2, \cdots, i_t\}$.*

2) *If $D = \langle \frac{g_i^{(6,n)}(x) g_{i+1}^{(6,n)}(x) g_{i+2}^{(6,n)}(x)}{\prod_{i\in B} M_i(x)} \rangle$, then $D_i \subset D$, where B is some nonempty subset of $\{i_1, i_2, \cdots, i_t\} \cup \{(i+1)_1, (i+1)_2, \cdots, (i+1)_t\} \cup \{(i+2)_1, (i+2)_2, \cdots, (i+2)_t\}$.*

Proof. The results come from the definition of dual-containing codes. □

4 Quantum Synchronizable Codes from the Obtained Cyclic Codes

First we review some basic concepts of QSCs. An $[[n, k]]$ quantum error-correcting code encodes k logical qubits into n physical qubits. A QSC with parameters (c_l, c_r)-$[[n, k]]$ is an encode scheme that corrects not only bit errors and phase errors but also a misalignment up to the left by c_l qubits and up to the right by c_r qubits, where c_l and c_r are nonnegative integers.

We provide the construction of QSCs by applying the method discovered by Fujiwara *et al.* [4, 6].

Lemma 6. *[4] Let $C_1 = \langle g_1(x) \rangle$ and $C_2 = \langle g_2(x) \rangle$ be two cyclic codes of parameters $[n, k_1, d_1]_r$ and $[n, k_2, d_2]_r$ respectively in $\mathbb{F}_r$ with $k_1 > k_2$ such that $C_2 \subset C_1$ and $C_2^{\perp} \subseteq C_2$. Define $f(x) = \frac{g_2(x)}{g_1(x)}$ in $\mathbb{F}_r[x]/(x^n-1)$. Then for any pair of nonnegative integers c_l, c_r satisfying $c_l + c_r < \mathrm{ord}(f(x))$, we can construct a (c_l, c_r)-$[[n+c_l+c_r, 2k_2-n]]$ QSC from C_1 and C_2 that can correct up to $\lfloor \frac{d_1-1}{2} \rfloor$ bit errors and $\lfloor \frac{d_2-1}{2} \rfloor$ phase errors.*

4.1 Maximum Misalignment Tolerance

Lemma 7. *The tolerable magnitude of QSCs is upper bounded by its length n.*

Proof. From Lemma 6, the synchronization capability of QSCs is related to the order of $f(x)$. According to the definition of $f(x)$ and $f(x)|(x^n-1)$, it is clear that the tolerable magnitude of QSCs is upper bounded by its length n. □

Lemma 8. *Let $n = 12m + 7$ be an odd prime. Then the tolerable magnitude of QSCs with length n can reach the upper bound.*

Proof. As the order of η is n in $\mathbb{F}_{q^{\mathrm{ord}_n(q)}}$, where n is an odd prime. We know that the order of any root of $f(x)$ is n, then the order of $f(x)$ must be n. By Lemma 7, the tolerable magnitude of QSCs constructed by Lemma 6 can reach the upper bound n. □

Based on the cyclic code C_i constructed in Lemma 2, we can obtain a class of QSCs as follows, whose synchronization capabilities can always reach the upper bound.

Theorem 3. *Let $n = 12m + 7$ be an odd prime, $t = \frac{n-1}{6\ell}$ and $q \in C_0^{(6,n)}$, where $q^{\ell} \equiv 1 \bmod n$. For any nonnegative integers c_l and c_r satisfying $c_l + c_r < n$, we can construct a*

QSC with parameters (c_l, c_r)-$[[n+c_l+c_r, 2|A|\ell+\frac{2n+1}{3}]]_q$, *where* $|A|$ *is the size of* A *in Lemma 5, and* $0 \le |A| \le t-2 = \frac{n-12\ell-1}{6\ell}$.

Proof. From the definition of cyclotomic coset, we have that the size of $C_{(s,n)}$ is ℓ and $\ell|(2m+1)$, for any $s \in \{1, 2, \ldots, n-1\}$. It is obvious that the cyclic code $C_i = \langle g_i^{(6,n)}(x)\rangle$ has augmented codes if and only if $g_i^{(6,n)}(x)$ has at least $t = \frac{n-1}{6\ell}$ irreducible factors in $\mathbb{F}_q[x]$. Since the size of $C_i^{(6,n)}$ is odd, we let $t \ge 3$. According to (a) in Lemma 5, the cyclic code $C_i^{(6,n)} = \langle g_i^{(6,n)}(x)\rangle$ has an augmented code C with parameters $\left[n, \frac{5n+1}{6} + |A|\ell\right]$. Taking a set A' such that $A \subset A' \subset \{i_1, i_2, \ldots, i_t\}$, then we can get a cyclic code C' with parameters $\left[n, \frac{5n+1}{6} + |A'|\ell\right]$ such that $C \subset C'$. Then $0 \le |A| \le t-2 = \frac{n-12\ell-1}{6\ell}$. Furthermore, by Lemmas 7 and 8, we can obtain a QSC with parameters (c_l, c_r)-$[[n+c_l+c_r, 2|A|\ell+\frac{2n+1}{3}]]_q$, whose tolerable magnitude against misalignment errors can reach the upper bound n. □

Moreover, we can also construct another class of QSCs whose synchronization capabilities reach the upper bound by using the cyclic code D_i and its augmented codes.

Theorem 4. *Let* $n = 12m+7$ *be an odd prime,* $t = \frac{n-1}{6\ell}$ *and* $q \in C_0^{(6,n)}$, *where* $q^\ell \equiv 1 \bmod n$. *For any nonnegative integers* c_l *and* c_r *satisfying* $c_l + c_r < n$, *we can construct a QSC with parameters* (c_l, c_r)-$[[n+c_l+c_r, 2|B|\ell+1]]_q$, *where* $|B|$ *is the size of* B *in Lemma 5, and* $0 \le |B| \le 3t-2 = \frac{n-4\ell-1}{2\ell}$.

Proof. Since the size of $C_{(s,n)}$ is ℓ and $\ell|(2m+1)$, it is obvious that the cyclic code $D_i = \langle g_i^{(6,n)}(x) g_{i+1}^{(6,n)}(x) g_{i+2}^{(6,n)}(x)\rangle$ has augmented codes if and only if $g_{i+j}^{(6,n)}(x)$ $(j \in \{0,1,2\})$ has at least $t = \frac{n-1}{6\ell}$ irreducible factors over $\mathbb{F}_q$. So we let $t \ge 3$. According to (b) in Lemma 5, the cyclic code $D_i = \langle g_i^{(6,n)}(x) g_{i+1}^{(6,n)}(x) g_{i+2}^{(6,n)}(x)\rangle$ has an augmented code D with parameters $\left[n, \frac{n+1}{2} + |B|\ell\right]$. Taking a set B' such that $B \subset B' \subset (\{i_1, i_2, \ldots, i_t\} \cup \{(i+1)_1, (i+1)_2, \ldots, (i+1)_t\} \cup \{(i+2)_1, (i+2)_2, \ldots, (i+2)_t\})$, then we can get a cyclic code $D' = \left[n, \frac{n+1}{2} + |B'|\ell\right]$ such that $D \subset D'$. Then $0 \le |B| \le 3t-2 = \frac{n-4\ell-1}{2\ell}$. Furthermore, by Lemmas 7 and 8, we can obtain a QSC with parameters (c_l, c_r)-$[[n+c_l+c_r, 2|B|\ell+1]]_q$ whose tolerable magnitude against misalignment errors can reach the upper bound n. □

The following are two examples about QSCs which are constructed by sextic cyclotomy. In particular, we can give a lower bound of the error-correcting capability towards bit errors and phase errors of QSCs constrtucted by D_i and its augmented codes.

Example 3. (a) *Let* $n = 127$ *and* $q = 19 \in C_0^{(6,n)}$. *In this case, we only consider the construction of QSCs from the cyclic code* $C_0 = \langle g_0^{(6,127)}(x)\rangle$ *and its augmented codes. Then* $0 \le |A| \le 5$, *by Theorem 3. From Example 2, let* $A = \{8, 16, 32, 64\}$, $|A| = 4$. *Then the cyclic code* $C = \langle \frac{g_0^{(6,127)}(x)}{\prod_{i\in A} M_i(x)}\rangle$ *with parameters* $[127, 118, 6]_{19}$ *is optimal and* $C^\perp \subset C$. *Furthermore, let* $A \subset A' = \{2, 4, 8, 16, 32, 64\}$. *Then the cyclic code* $C' = \langle \frac{g_0^{(6,127)}(x)}{\prod_{i\in A'} M_i(x)}\rangle$ *with parameters* $[127, 124, 3]_{19}$ *is optimal and* $C^\perp \subset C \subset C'$. *Then by Lemma 6, we can construct a* (c_l, c_r)-$[[127+c_l+c_r, 109]]_{19}$ *QSC with* $c_l + c_r < 127$, *whose tolerable magnitude against misalignment errors can reach the upper bound. Moreover, since the cyclic codes* C *and* C' *are optimal, the QSC we construct has the optimal error-correcting capability towards bit errors and phase errors.*

(b) *Let the notations be defned as above. In this case, we only consider the QSCs constructed from* $D_0 = \langle g_0^{(6,127)}(x) g_1^{(6,127)}(x) g_2^{(6,127)}(x)\rangle$ *and its augmented codes. By Theorem 4,* $0 \le |B| \le 19$. *From Example 2, let* $B = \{2\}$. *Hence the augmented code of* D_0 *is* $D = \langle \frac{g_0^{(6,127)}(x) g_1^{(6,127)}(x) g_2^{(6,127)}(x)}{\prod_{i\in B} M_i(x)}\rangle$. *By Lemma 3,* D_0 *is a dual-containing code, then we have* $D_0^\perp \subset D_0 \subset D$. *From Lemma 6, we can construct a* (c_l, c_r)-$[[127+c_l+c_r, 1]]_{19}$ *QSC with* $c_l + c_r < 127$, *whose tolerable magnitude against misalignment errors can reach the upper bound. From Theorem 1, the lower bound of* D_0 *satisfies* $d_0^2 - d_0 + 1 \ge 127$, *then the parameters of* D_0 *are* $[127, 64, \ge 12]_{19}$ *and the parameter of* D *are* $[127, 67, \ge 12]_{19}$. *According to Lemma 6, the QSCs we constructed can correct up to 5 bit errors and 5 phase errors.*

5 Conclusion

We study two classes of QSCs from dual-containing cyclic codes obtained by sextic cyclotomic classes. The constructed QSCs possess the highest tolerance against misalignment errors, besides some of them have optimal or almost optimal error-correcting capability towards bit errors and phase errors. Since the exact Hamming distances of the cyclic codes used to construct QSCs are quite difficult to compute, the error-correcting capability of QSCs is difficult to determine in theory. We hope that our future work can make a breakthrough in this respect.

Acknowledgments

This work was supported by Fundamental Research Funds for the Central Universities (20CX05012A), the Major Scientific and Technological Projects of CNPC under Grant(ZD2019-183-008), National Nature Science Foundation of China (11775306), and Shandong Provincial Natural Science Foundation of China (ZR2019MF070).

References

[1] W. Bosma, J. J. Cannon, and C. Fieker, "Handbook of magma functions," *Journal of Symbolic Computation*, vol. 24, no. 3-4, p. 5017, 2010.

[2] A. E. Brouwer, *Bounds on the size of linear codes*. Elsevier, 1998.
[3] A. R. Calderbank and P. W. Shor, "Good quantum error-correcting codes exist," *Physical Review A*, vol. 54, no. 2, pp. 1098–1105, 1996.
[4] Y. Fujiwara, "Block synchronization for quantum information," *Physical Review A*, vol. 87, no. 2, p. 022344, 2013.
[5] Y. Fujiwara, V. D. Tonchev, and T. Wong, "Algebraic techniques in designing quantum synchronizable codes," *Physical Review A*, vol. 88, no. 1, p. 012318, 2013.
[6] Y. Fujiwara and P. Vandendriessche, "Quantum synchronizable codes from finite geometries," *IEEE Transactions on Information Theory*, vol. 60, no. 11, pp. 7345–7354, 2014.
[7] W. Gao and T. Yan, "Double circulant self-dual codes from generalized cyclotomic classes of order two," *International Journal of Network Security*, vol. 23, no. 3, pp. 395–400, 2021.
[8] M. Grassl, "Bounds on the minimum distance of linear codes and quantum codes," Mar. 2, 2022. (`http://www.codetables.de`)
[9] G. G. L. Guardia, *Quantum Error Correction: Symmetric, Asymmetric, Synchronizable, and Convolutional Codes*. Switzerland: Springer International Publishing, 2020.
[10] W. C. Huffman and V. Pless, *Fundamentals of Error-Correcting Codes*. Cambridge University Press, 2003.
[11] C. Jiang, Y. Sun, and X. Liang, "Eight power residue double circulant self-dual codes," *International Journal of Network Security*, vol. 22, no. 5, pp. 736–742, 2020.
[12] L. Q. Li, S. X. Zhu, and L. Liu, "Quantum synchronizable codes from the cyclotomy of order four," *IEEE Communications Letters*, vol. 23, no. 1, pp. 12–15, 2019.
[13] X. Li and Q. Yue, "A new family of quantum synchronizable codes," *IEEE Communications Letters*, vol. 25, no. 2, pp. 342–345, 2021.
[14] L. Luo and Z. Ma, "Non-binary quantum synchronizable codes from repeated-root cyclic codes," *IEEE Transactions on Information Theory*, vol. 64, no. 3, pp. 1461–1470, 2018.
[15] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. New York: Elsevier, 1977.
[16] M. A. Nielsen and I. Chuang, *Quantum Computaion and Quantum Information*. New York: Cambridge University Press, 2011.
[17] A. M. Steane, "Error correcting codes in quantum theory," *Physical Review Letters*, vol. 77, no. 5, pp. 793–797, 1996.
[18] Y. X. Xie, Yuan J. H, and Y. Fujiwara, "Quantum synchronizable codes from quadratic residue codes and their supercodes," in *2014 IEEE Information Theory Workshop (ITW 2014)*, pp. 172–176, August 2014.

Biography

Tao Wang received the B.S. degree in China University of Petroleum, Qingdao China, in 2018. He is currently pursuing his M.S. in Mathematics from China University of Petroleum. His research interests include coding theory and Information security.

Xueting Wang received the B.S. degree in University of Jinan, Jinan China, in 2020. She is currently pursuing her M.S. in Mathematics from China University of Petroleum. His research interests include coding theory and Information security.

Qian Liu is a undergraduate student of China University of Petroleum. He is mainly engaged in the research of Applied mathematics.

Tongjiang Yan was born in 1973 in Shandong Province of China. He was graduated from the Department of Mathematics, Huaibei Coal-industry Teachers College, China, in 1996. In 1999, he received the M. S. degree in mathematics from the Northeast Normal University, Lanzhou. He received the Ph. D. degree in cryptography from the Xidian University, Xian. He is now a professor of China University of Petroleum. His research interests include cryptography and coding.

Adaptive Intrusion Detection Model Based on CNN and C5.0 Classifier

Wen-Tao Hao[1], Ye Lu[2], Rui-Hong Dong[3], Yong-Li Shui[3], and Qiu-Yu Zhang[3]
(Corresponding author: Wen-Tao Hao)

Network Information Center of Xi'an Aeronautical University[1]
No.259, West-Second-Ring Road, Xian 710077, China
Email: haowentao811@163.com
School of Computer Science, Baoji University of Arts and Sciences[2]
No. 1, Gaoxin Avenue, Baoji City 721013, China
School of Computer and Communication, Lanzhou University of Technology[3]
No.287, Lan-Gong-Ping Road, Lanzhou 730050, China
(Received Jan. 7, 2022; Revised and Accepted Apr. 28, 2022; First Online May 1, 2022)

Abstract

In order to solve the problems of traditional intrusion detection methods in industrial control networks that are difficult to adaptively respond to dynamic changes in the network environment, extract valid data features, and low detection rates for unknown attacks, an adaptive intrusion detection model based on convolutional neural network (CNN) and C5.0 classifier was proposed. The proposed model first uses the synthetic minority oversampling method (SMOTE) to solve the problem of data type imbalance. Then the middle hidden layer of CNN is used to realize the automatic extraction of network traffic data features. Finally, the C5.0 classifier model is trained using the training set data extracted from CNN. An adaptive online update strategy based on frequent pattern mining is introduced so that the intrusion detection model can adapt to the dynamic changes of the network environment and then obtain the final detection result. The experiment uses KDDCup 99, NSL-KDD, and Gas Pipeline datasets to test the model's validity. Experimental results show that compared with the existing methods, the proposed model can effectively adapt to the dynamic changes of the network environment, and the classification and detection accuracy of various attack behaviors can reach over 98%. In addition, the false alarm rate is less than 2%.

Keywords: Adaptive Intrusion Detection; C5.0 Decision Tree; Convolutional Neural Network; Industrial Control Network; Synthetic Minority Oversampling

1 Introduction

With the continuous integration development of industrialization and informatization, more and more researchers pay attention to the field of network security of industrial control systems. In the industrial control network environment, while responding to traditional functional security threats, industrial control systems are also facing more and more industrial control information security threats such as viruses, Trojan horses, and hackers [7]. There are some malware (such as Stuxnet in 2010, BlackEnergy in 2013, EternalBlue in 2017, etc.) that sounded the alarm for the information security of industrial control networks [26]. Therefore, research on the security of industrial control networks has very important research significance [1].

In view of the security problems of industrial control networks, traditional industrial control security technologies, such as user authentication, firewall, and data encryption, can no longer cope. As the second line of defense of industrial control network information security, intrusion detection technology can effectively make up for the shortcomings of traditional industrial control security technology [24]. Intrusion detection technology extracts data characteristics that reflect system behavior, and classifies attack behavior and normal behavior data through a designed detection algorithm. Due to the continuous improvement of network intrusion technologies and methods, no matter whether it is misuse detection or anomaly detection, satisfactory results can not be obtained [3]. In recent years, scholars have combined intrusion detection technology with the current mainstream algorithms to extend new research directions. They have successively applied mature intrusion detection technologies such as machine learning, data mining, deep learning and so on to the intrusion detection of industrial control networks, and constantly innovating, especially machine learning techniques, such as support vector machines, Bayesian networks, decision trees and other methods [18] have achieved good results. However, due to the

diversification of network intrusions and the imbalance of intrusion data classes, intrusion detection technology cannot detect some new or unknown forms of attacks, cannot adapt to the dynamic changes of the network environment, and cannot achieve the global scope of intrusion detection functions, resulting in low detection efficiency [6, 15, 22]. In addition, with the increase of heterogeneous networks, the network environment has become more and more complex, and the attack behaviors and methods of intruders are constantly evolving and updating. Because of the continuous appearance of unknown intrusions and the dynamic changes of the network environment, the detection rate of the intrusion detection model may be low and the false alarm rate is high. The adaptive intrusion detection technology is an important mechanism for the dynamic changes of the network environment. Which can make the intrusion detection model adapt to the dynamic changes of the network environment and improve the detection rate [19].

In order to reduce the influence of the dynamic changes of the network environment on the accuracy of the intrusion detection model, to better extract effective data features and improve the detection performance of the intrusion detection model, we presents an adaptive intrusion detection model based on CNN and C5.0. The proposed model first preprocesses the dataset. Then, the pooling layer in CNN is used to transform one-dimensional data into multidimensional data, and the optimal features are selected by CNN adaptive. Finally, the optimal features selected by CNN is used to train and detect C5.0 classifier, and an adaptive online update strategy of frequent pattern mining is introduced in the detection process, so that the intrusion detection model can adapt to the dynamic changes of the network environment. The main contributions of this paper are as follows:

1) The hidden layer of CNN is used to realize automatic extraction of network traffic data features. In the process of feature extraction, synthetic minority oversampling is used to solve the problem of data class imbalance.

2) Combining CNN with C5.0 classifier realizes the optimal classification of normal and attack behavior in the intrusion detection model.

3) In the C5.0 classification and detection process, an adaptive online update strategy based on frequent pattern mining is introduced, so that the intrusion detection model can adapt to the dynamic changes of the network environment, and has a high detection rate for known and unknown attacks.

The remaining part of this paper is organized as follows. Section 2 introduces related work. Section 3 introduces related theories in detail. Section 4 describes the proposed adaptive intrusion detection model and related methods. Section 5 gives the experimental results and performance analysis as compared with existing methods. Finally, we conclude our paper in Section 6.

2 Related Work

The dynamic changes of the network environment and the detection of unknown attacks are the main challenges faced by intrusion detection models in current industrial control networks. Therefore, the intrusion detection model that can adapt to the dynamic changes of the network environment and the detection of new intrusion attacks is called an adaptive intrusion detection model. Many researchers at home and abroad have applied mature intrusion detection technologies such as data mining, machine learning, and deep learning into the research of the adaptive intrusion detection model.

Combining data mining with adaptive intrusion detection technology can dig out the deep features of the data, so that intrusion detection has a certain adaptive ability. For example, Ref. [17] used data mining methods to detect abnormal behaviors in incoming datasets, and proposes an intrusion detection system based on self-learning technology. Ref. [21] proposed a knowledge-based intrusion detection strategy for current wireless sensor networks, which is used to detect various forms of attacks under different network structures. Ref. [13] proposed an adaptive network intrusion detection method based on fuzzy rough set feature selection and GA-GOGMM model learning, which can effectively adapt to the dynamic changes of the network environment and can detect various intrusion behaviors in real network connection data in real time.

At present, there have been many related works applying machine learning methods to adaptive intrusion detection. Such as, Ref. [19] proposed an incremental machine learning classifier for intelligent detection and analysis of network data streams. This is an adaptive intrusion detection model (AIDM) based on machine learning technology and feature extraction, which can detect unknowns attack. In order to detect unknown attacks in real-time network traffic, Ref. [2] proposed an adaptive intrusion detection system using multi-layer hybrid support vector machine and extreme learning machine technology to detect and learn unknown attacks in real time. Setareh Roshan et al. [23] proposed an adaptive design method of intrusion detection system based on extreme learning machine, which improved the rapid learning and real-time detection capabilities of intrusion detection system. Ref. [14] proposed an adaptive network intrusion detection (ANID) method based on kernel extreme learning (KELMs) random feature selection integration. This method updates the intrusion detection model according to the dynamic changes of the network environment, guarantees the adaptive ability of the intrusion detection model, and achieves high detection accuracy for both known and unknown attacks, and improves the detection efficiency. In [25], in response to the class imbalance problem in the intrusion data set, the synthetic minority oversampling technique (SMOTE) is used to balance the data set, and then the random forest training classifier is used for intrusion detection.

In recent years, deep learning has been widely used in

the field of intrusion detection and has certain advantages. Ref. [20] proposed an adaptive misuse intrusion detection system combining self-learning and APE-K framework, which can detect unknown attacks by using deep learn-based methods in network environment changes. Chu Ankang et al. [4] proposed an industrial controlled intrusion detection method based on multi-classification long-short memory model, which has good detection accuracy, but cannot accurately detect unknown attacks in high-dimensional and complex changing network environments. Ref. [9] proposed an industrial control detection scheme based on deep learning methods. Deep learning methods can automatically extract key features to achieve accurate attack classification. Ref. [10] proposed an intrusion detection method using multi-CNN fusion method for deep learning, which fully contributes to the data of the Industrial Internet of Things. In [28], for the problem of data class imbalance,put forward the technique of combining SMOTE and the Gaussian mixture model (GMM), and unbalanced processing is combined with a convolutional neural network. Ref. [11] proposed a new feature-level IDS based on convolutional neural networks, and achieved good performance.

Through the above analysis, it can be seen that both data mining and traditional machine learning methods can effectively improve the detection rate, but both have weak adaptability to dynamic changes in the network environment and low detection rate for unknown attacks. Intrusion detection based on deep learning has the advantage of detecting unknown attacks, and can automatically identify different attack characteristics, so as to find potential security threats more efficiently. Therefore, in view of the weak adaptive ability of intrusion detection technology in industrial control networks to dynamic changes in the network environment, difficulty in extracting valid data features, and low detection rate of unknown attacks, this paper proposes adaptive intrusion detection model based on CNN and C5.0 classifiers. The model uses CNN to automatically select the features of the dataset, then uses the C5.0 classifier for training and detection, and introduces an adaptive update strategy for frequent pattern mining.

3 Related Theories

3.1 Convolutional Neural Network Model

CNN [8] generally consists of a convolutional layer part and a fully connected layer part, where the number of convolutional layers and pooling layers of CNNs with different structures is different. CNN uses a back-propagation learning process, that is, input training data in the input layer. Then the predicted value is calculated through the calculation of the convolutional layer, the pooling layer, the fully connected layer and the output layer. Finally, the error function is used to calculate the difference between the real value and the predicted value, so as to achieve the purpose of reverse iteration to update the network weights and thresholds. The structure of CNN is shown in Figure 1.

Convolutional layer: The convolutional layer is the core part of CNN. The feature extraction module in CNN is composed of a combination of multiple convolutional layers. In the convolutional layer, the input feature map and the convolution kernel are convolved and biased, and then a non-linear activation function is used to obtain the feature map of the new layer. The current convolution feature is obtained by the convolution operation of the convolution kernel and the output feature of the previous layer. Its definition is shown in Equation (1):

$$Y_j^a = \sum_i X_i^{a-1} * T_{ij}^{a-1} + b_j^a \tag{1}$$

where Y_j^a represents the input of the jth position in the ath layer feature after the convolution operation, X_i^{a-1} represents the ith input matrix in the a-1 layer, T_{ij}^{a-1} represents the convolution kernel connecting the ith input matrix and the jth position between the ath layer and the a-1th layer, and b_j^a is the offset of the jth position in the features of the ath layer.

After the calculation of Equation (1), the obtained matrix needs to undergo nonlinear activation, which can strengthen the nonlinear expression ability of the network. Commonly used activation functions are Sigmoid, Relu, etc. Pooling layer: The pooling layer is sampled under the output feature map of the convolutional layer. Under the premise of retaining the main features of the data, the data features are reduced in dimensionality, which increases the generalization ability of the neural network and achieves the removal of redundancy. The effect of this also reduces the complexity of the entire network.

Fully connected layer: The fully connected layer acts as a classifier in the entire CNN, that is, after convolution, activation function, pooling and other deep networks, the results are identified and classified through the fully connected layer. The calculation formula is shown in Equation (2):

$$y_j^a = \sum_i K_{ij}^a * x_i^{a-1} + b_j^a \tag{2}$$

where y_j^a is the calculated output result of the jth neuron in the fully connected layer t, K_{ij}^a represents the connection weight value of the ith feature in the a-1th layer and the jth neuron in the ath layer, x_i^{a-1} represents the ith eigenvalue of the features of the a-1 layer, which is the offset of the jth neuron in the fully connected layer a.

After the feature data passing through the convolutional layer and pooling layer pass through the last classification prediction layer, the classification and prediction results can be obtained.

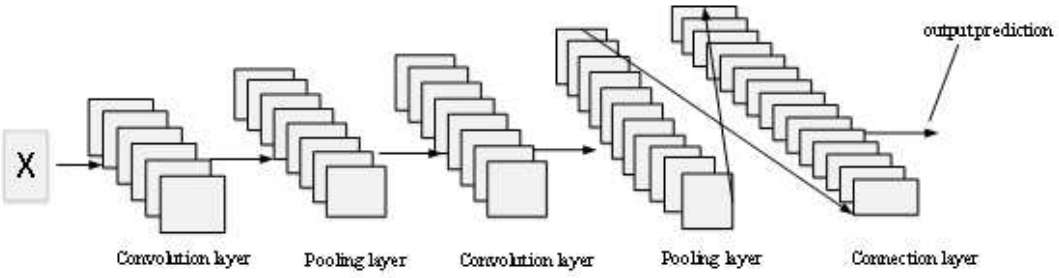


Figure 1: Structure diagram of CNN

3.2 Synthetic Minority Oversampling (SMOTE)

In order to solve the problem of data class imbalance, a synthetic minority oversampling is used to preprocess the training set data. Synthetic minority oversampling [24] is an improved scheme based on random oversampling, because random oversampling is prone to the problem of model overfitting, which makes the information learned by the model too special and not general enough. Synthetic minority oversampling changes the data distribution of the unbalanced dataset by adding the generated minority samples, and synthesizes new samples between two minority samples by linear interpolation, thereby effectively alleviating the overfitting caused by random oversampling problem. The specific process of synthesizing minority oversampling is:

Step 1. Calculate the K-nearest neighbor sample set of each sample V_i of the minority class in the training set.

Step 2. Analyze the proportion of the majority in the sample set, and then judge whether V_i is a boundary sample, if it is, add the boundary sample set; otherwise, put it back into the minority sample set.

Step 3. The boundary sample V_i is oversampled to generate a new minority sample V_new, whose definition is shown in Equation (3):

$$V_{new} = V_i + rand(0,1) \times \mid V_j - V_i \mid \tag{3}$$

where j=1,2,...,n, n represents the number of samples selected randomly according to the sampling ratio, and $rand$(0,1) represents the random number [0,1].

3.3 C5.0 Decision Tree

The C5.0 decision tree [27] classification algorithm is a new classification algorithm that is improved on the basis of the C4.5 classification algorithm. The decision tree construction idea of C5.0 algorithm is consistent with that of C4.5 algorithm, and C5.0 algorithm also includes all the functions of C4.5 algorithm. The difference from the C4.5 algorithm is that the C5.0 algorithm introduces Boosting technology and cost matrix construction technology.

Compared with the C4.5 algorithm, the improvements of the C5.0 decision tree algorithm are: C5.0 runs much faster than the C4.5 decision tree algorithm, C5.0 usually uses less memory than C4.5, and the number of C5.0 trees is less than C4.5.

4 The Proposed Model

4.1 Adaptive Intrusion Detection Model Based on CNN and C5.0

Aiming at the problem of the dynamic changes of the network environment in the industrial control system, in order to improve the adaptability of the intrusion detection model in the dynamic environment, this paper uses a series of algorithms for synthesizing minority oversampling, CNNs, C5.0 decision trees and frequent pattern mining combined, an adaptive intrusion detection model is designed. The model is mainly composed of data preprocessing, classification representation, matching update and attack response. Figure 2 shows the structure of an adaptive intrusion detection model based on CNN and C5.0 classifier.

It can be seen from Figure 2 that the model first uses the hidden layer of the CNN to realize the automatic selection of data features and reduce the dimension of data features. Then use the training set data to train the C5.0 classifier model to obtain the classification of the normal library and the abnormal library. Finally, an adaptive matching update mechanism is introduced, by introducing frequent pattern mining, real-time matching and updating the normal database and abnormal database, to ensure the adaptability of the model, and realize the detection of various attack behaviors, thus improving the network intrusion detection model performance.

The detailed processing steps of the adaptive intrusion detection model are as follows:

Step 1. Data preprocessing. Firstly, the original dataset is preprocessed, the digitized character type in the dataset is integer, and the attribute feature value is normalized. For the problem of data imbalance in the training dataset, a synthetic minority oversampling method is used to deal with it.

Step 2. Feature selection. CNN is used to automatically

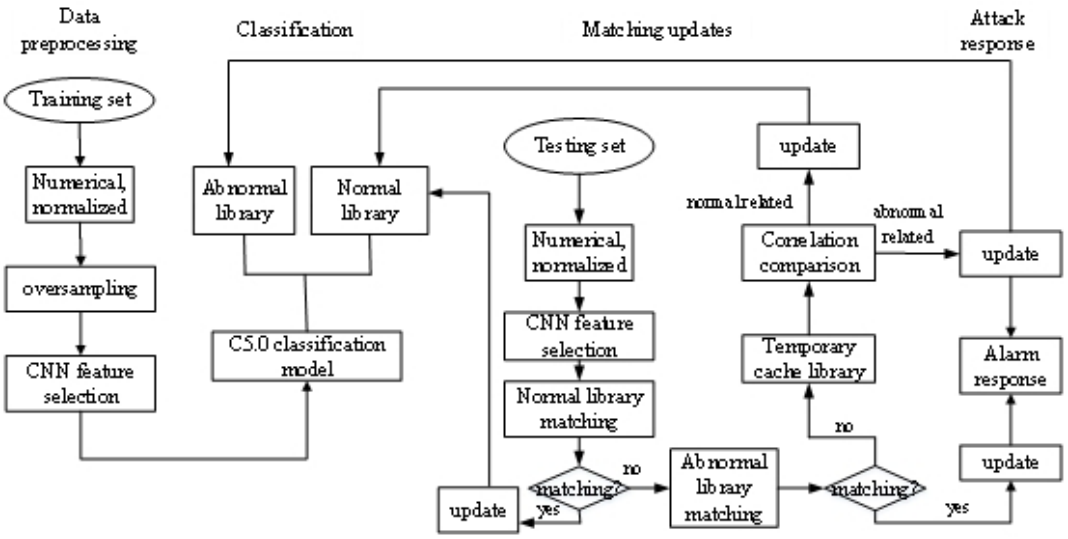


Figure 2: Flow chart of adaptive intrusion detection model processing based on CNN and C5.0 classifier

select data features of the preprocessed data and reduce the dimension of data features.

Step 3. The classification database is formed, and the C5.0 classifier model is trained using the training set data to obtain the normal library and the abnormal library.

Step 4. Matching and update, in view of the dynamic changes of the network environment, in order to improve the adaptability of the intrusion detection model in the dynamic environment, an adaptive matching update mechanism is introduced. By building a temporary cache library, the data that does not match the normal library and the intrusion library is added to the temporary cache library.

A. The test set data M is searching and matching with the normal database. If a type that matches M is found in the normal library, update the data, mark M as normal, and re-execute this step to detect new samples; if it does not match, execute step B.

B. Search and match M in the exception library. If a type that matches M is found, the alarm responds, and the update is performed at the same time, ending the processing of this record, and returning to step A to test the new sample; if it does not match, proceed to the next step.

C. Update the cosine similarity by comparing the cosine similarity with normal and abnormal in the temporary cache library, and mark the high similarity with normal as normal, otherwise, mark as abnormal.

Step 5. Attack response, through the matching update, and the intrusion behavior matching the abnormal library for alarm response. The behaviors that are highly related to intrusions in the temporary cache library also respond to alarms.

4.2 CNN Construction

The CNN network can use the hidden layer to learn the local features of the data layer by layer, and extract the data features in the spatial dimension. The preprocessed network traffic data features are input into the CNN network, and the CNN features are output from the output layer after layer-by-layer learning. Figure 3 shows the CNN model used in this paper.

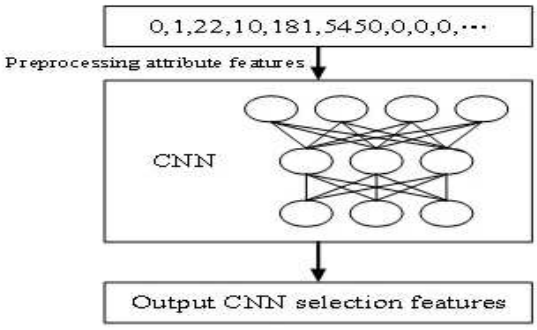


Figure 3: CNN model diagram

4.3 C5.0 Classifier Training

In the C5.0 classifier algorithm, Boosting technology usually stacks multiple C4.5 weak classifiers into one strong classifier. The algorithm inputs the training dataset into the classifier, and trains the weak classifiers one by one in the order of a ladder-like training process. Each time the training set of the weak classifier is transformed according to a certain strategy, and finally the weak classifier is combined into a strong classifier in a certain way.

In the proposed model, the Boosting algorithm is added to the C5.0 algorithm through the C5.0 function. In the C5.0 function, the role of the trials parameter is to control the number of C4.5 decision trees and to enhance the classification performance of the C5.0 model. After building the C5.0 classifier, start training the C5.0 classifier. By adjusting the specific values of parameters such as trials, the model is optimized, so that the model can achieve better classification performance. Figure 4 is a flow chart of C5.0 classifier construction and classification.

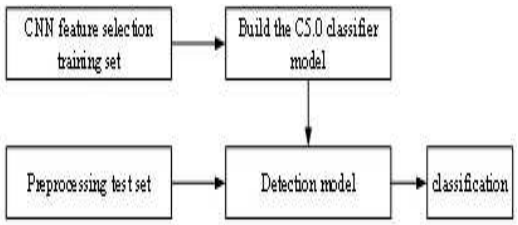


Figure 4: C5.0 classifier construction and classification flow chart

4.4 Adaptive Matching Update Strategy

In order to improve the weak adaptability of the intrusion detection model to the dynamic changes of the network environment, this paper introduces an adaptive matching update strategy. By building a temporary cache library, data that does not match the normal library and the abnormal library is added to the temporary cache library. Each type of data in the temporary cache library has a value, and when a new record matches this type, the value of the type is increased by 1. By analyzing the similarity between the type and the normal type or abnormal type, it is determined to add this type to the normal or abnormal library for dynamic matching and updating. In matching and updating, the cosine similarity is used to measure the similarity between the object to be detected and the corresponding type for matching and updating. The calculation method of cosine similarity is as Equation (4):

$$d(x, y) = x^T y / \| x \| \| y \| \quad (4)$$

where $d(x, y)$ represents the cosine matching degree between the sample x and the type y to be detected. The larger the value, the higher the matching degree (representing the smaller the angle between x and y).

Figure 5 is a flow chart of adaptive matching update strategy processing.

5 Experimental Results and Analysis

The experimental environment of this paper is Intel Core i5-4210U 2.49ghz,8G,Windows 10(64-bit). The programming language is python3 and R, deep learning uses the Keras deep learning framework based on Tensorflow, and data preprocessing uses weka 3.8.3. The experimental dataset selected three datasets: KDDCup 99 [16], NSL-KDD [5] and Gas Pipeline [12].

In the feature selection experiment, in order to prevent the imbalance of data types from causing more false positives, a few synthetic oversampling methods are used for data preprocessing, and then the hidden layer in CNN is used for data feature selection, setting the Dropout rate to 0.5 and hiding in the middle The layer activation function is Relu, and use Root Mean Square Prop (RMSProp) as the optimization function. Since one-hot encoding is not used when digitizing attributes and labels, and numerical encoding is used directly, the model in this paper uses Sparse Categori-Calcrossentropy as the loss function of the network. The initial learning rate of the model is 0.0001. In the classification experiment, for the C5.0 classifier model, set the trial parameter value to 10, and select the default parameters for other parameters. In the model feature learning stage, the most representational feature subset is extracted by constantly adjusting the value of the intermediate feature extraction layer. The adaptive matching updating mechanism is used to integrate the matching updating classification of test set data, judge whether it is abnormal or normal behavior, and divide it into correct classification. In addition, in order to prove the superiority of the intrusion detection performance of this method, the existing four classifiers of RF, SVM, C4.5 and KNN were compared.

5.1 Dataset Description

5.1.1 KDDCup 99 Dataset

KDDcup 99 [16] is one of the most widely used intrusion detection data sets. It contains 4.9 million attack records, which are divided into training sets and test sets. The training set contains one normal type and 22 attack types, while the test set contains another 17 unknown attacks. The included corrected is the test sample set, which includes 17 abnormal types that do not appear in the 10% training set to test the generalization ability of the model.

Each traffic record in the KDDcup 99 dataset consists of 41 feature attributes and 1 class label, containing three main types of features: Attributes 1-10 are the basic features of network connections, attributes 11-22 are the content features of network connections, and attributes 23-41 are the traffic features.

According to the features of the dataset attack, it is divided into the following four types of attacks: denial of service attacks (DoS), probe attacks (Probe), user to root attacks (U2R), root to local attacks (R2L). Some specific types of attacks only appear in the test set, which provides a more realistic theoretical basis for intrusion detection. The specific data information of the KDDcup 99 data set is shown in Table 1.

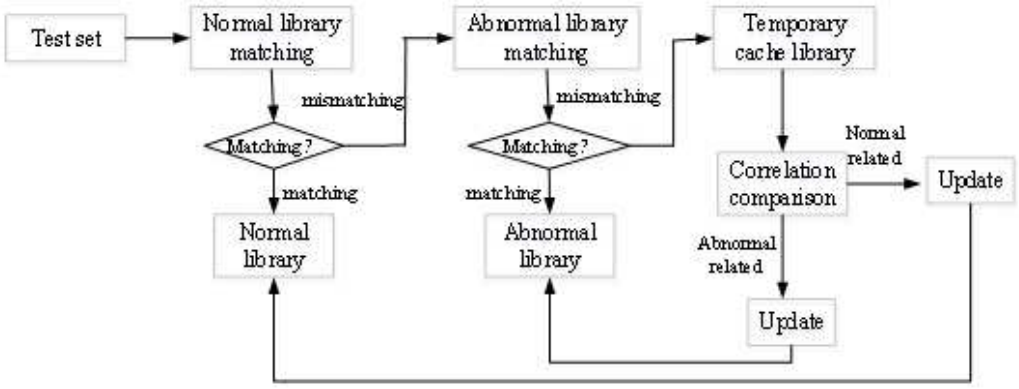


Figure 5: Adaptive matching update policy processing flow

Table 1: KDDcup 99 (10%) experimental dataset attack types and numbers

Classes	Training Data	Testing Data
Normal	97,278	60,593
Dos	391,458	22,985
Probe	4107	4166
R2L	1126	16,189
U2R	52	228

5.1.2 NSL-KDD dataset

The NSL-KDD dataset [5] is an improvement of the KDD99 dataset: (1) The training set of the NSL-KDD dataset does not contain redundant records, so the classifier will not be biased towards more frequent records; (2) There is no duplicate record in the test set of NSL-KDD dataset, which makes the detection rate more accurate; (3) The number of selected records from each difficulty level group is inversely proportional to the percentage of records from the original KDD dataset. The classification rates of different machine learning methods vary over a wider range, making accurate assessments of different learning techniques more effective; (4) The setting of the number of records in training and testing is reasonable, which makes the cost of running the experiment on the whole set of experiments low without having to randomly select a small part.

The specific data information of the NSL-KDD dataset is shown in Table 2.

Table 2: NSL-KDD dataset attack types and numbers

Classes	Training Data	Testing Data
Normal	67,343	9711
Dos	45,927	7458
Probe	11,656	2421
R2L	959	2754
U2R	52	200

5.1.3 Gas Pipeline

The industrial control system intrusion detection dataset Gas Pipeline [12] disclosed by Mississippi State University is a laboratory simulation-scale industrial control system network dataset based on Modbus application layer protocol. The dataset includes 1 type of normal data and 7 types of different attack data. Each record contains 26 traffic features and category labels. The dataset includes network traffic, process control and process measurement features. The dataset is captured by a network data logger, which monitors and stores Modbus traffic information from RS-232C connections, including normal, reconnaissance attack, and Response Injection (RI) attacks and Command Injection (CI) attacks and DoS attacks. This paper selects 80% of the natural gas pipeline dataset as the training set and 20% as the test set. The specific data information of the natural gas pipeline dataset is shown in Table 3.

Table 3: Types and numbers of attacks on the natural gas pipeline dataset

Classes	Training Data	Testing Data
Normal	86,137	128,443
Recon	1519	2268
NMRI	3079	4674
CMRI	5133	7902
MSCI	3044	4856
MPCI	8130	12282
MFCI	1951	2947
DoS	858	1318

5.2 Data Preprocessing

5.2.1 Data Oversampling

Due to the problem of data imbalance in the original dataset used in this paper, the learning algorithm will be biased towards records that appear more frequently. Therefore, this paper adopts the synthetic minority oversampling method to oversampling the minority boundary samples to make the synthesized sample distribution more

reasonable.

5.2.2 Data Transforming

The KDDcup 99 dataset and NSL-KDD dataset have 38 numerical characteristics and 3 non-numerical characteristics, such as protocol type, service, and label. The classification modules of the intrusion detection model all need to calculate the numerical flow features, so non-numerical features must be converted into numerical features. For instance, the protocol_type feature in the NSL-KDD dataset contains three types of protocols, namely TCP, UDP and ICMP, which are replaced by 1, 2, and 3 respectively. As well, the 70 service attributes and 11 flag attributes in the dataset are also numeralized in the same way. The character-type features in the Gas Pipeline dataset are also digitized in the same way.

5.2.3 Data normalization

Data normalization is a process of scaling the value of each attribute to a relatively good range, in order to eliminate the preference for features with larger values from the dataset. Since the KDDcup 99 dataset, NSL-KDD dataset and the Gas Pipeline dataset have data with no fixed upper and lower bounds and continuous values. Therefore, it is necessary to use min-max standardization to map the feature data to the standard range of [0, 1], and each feature is standardized using Equation (5):

$$f(x) = (x - min)/(max - min), x\epsilon[min, max] \quad (5)$$

where x is the feature in the dataset, min and max are the minimum and maximum of feature x. The normalized data can be directly input into the intrusion detection model.

5.3 Evaluation Indexes

To evaluate the performance of the intrusion detection model, Accuracy (Acc), Precision (Precision) andfalse alarm rate (FAR) [5] are used to measure the performance of the model. When evaluating the effectiveness of the model, most commonly used indicators can be calculated from the confusion matrix in Table 4.

Table 4: Confusion matrix

True value	Predictive value	
	Normal	Abnormal
Normal	TP	FN
Abnormal	FP	TN

In Table 4, TP indicates that the true value is a normal sample and is predicted to be the number of normal samples. FN indicates that the true value is a normal sample and is predicted to be the number of abnormal samples. FP indicates that the true value is an abnormal sample and is predicted to be the number of normal samples. TN means that the true value is an abnormal sample and is predicted to be the number of normal samples.

Accuracy (Acc): Accuracy represents the percentage of the dataset that the model correctly classifies the true value of the sample. When the various samples in the dataset are relatively average, this is a good measure. However, when the various types of samples is unbalanced, it cannot reflect the true classification effect of the model. The calculation formula is as follows:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad (6)$$

Precision: It represents the probability of actually being a positive sample among all the samples predicted to be positive. The calculation formula is as follows:

$$Precision = TP/(TP + FP) \quad (7)$$

False alarm rate: The false alarm rate is the ratio of the number of false positive records to the total number of normal records, reflecting the proportion of false alarm records in normal records. The formula is as follows:

$$FAR = FP/(FP + TN) \quad (8)$$

5.4 Performance Analysis

5.4.1 Influence of adaptive online update on the model

When adaptive online matching is updated, a frequent pattern threshold needs to be given to determine whether a new sample belongs to a frequent pattern, and then the pattern library is updated. In this paper, the mode attenuation parameter is 0.0001 in the experiment, and different frequent mode thresholds (100, 500, 1000) are selected at the same time to test the influence of different frequent mode thresholds on the model. The experimental results are shown in Figure 6.

As can be seen from Figure 6 that in the experiment, different frequent pattern thresholds have different influences on the accuracy and false alarm rate of the proposed model. The higher frequent threshold in the initial stage has a higher accuracy rate, with the increase of experimental data, the lower the frequent threshold, the higher the accuracy rate. Obviously, the smaller frequent threshold is faster than the adaptive online update strategy, which makes the normal library and abnormal library update more timely, and can ensure a higher intrusion detection accuracy rate. However, some fuzzy and frequent instances will be mistakenly introduced into the unmatched pattern library, which will lead to a high false alarm rate in the proposed model. Therefore, in order to obtain a higher detection rate and a lower false alarm rate, in the initial stage of the experiment, due to the fewer types of abnormal libraries and the large amount of data in the test set, the threshold of frequent patterns was first set to be larger. With the gradual increase in the types of the normal library and the abnormal library,

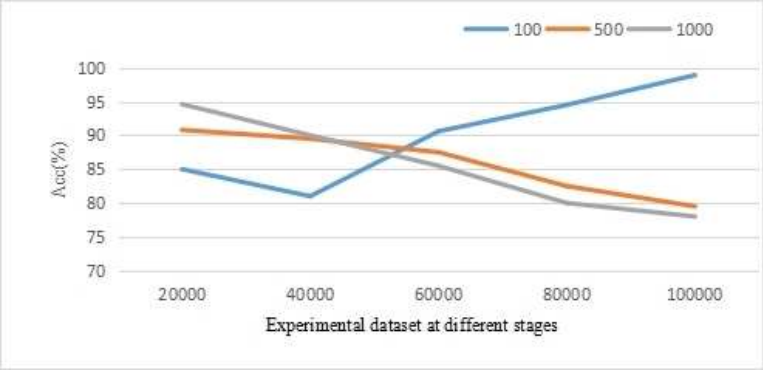


Figure 6: Experimental results of different frequent pattern mining thresholds

the correlation between most data and the normal library and the abnormal library is obviously increased, and the frequent pattern threshold can be automatically reduced according to the decrease in the amount of data in the temporary cache library.

5.4.2 Performance Comparison of CNN Feature + C5.0 Binary Classification Experiment

In order to evaluate the performance of the adaptive intrusion detection model combining the features of the intrusion detection dataset learned by CNN and the C5.0 classifier, ROC curve is used to represent the classification performance of C5.0, RF, SVM, C.4.5 and KNN. On the one hand, it can reflect the representativeness of the CNN selected features, on the other hand, it can also get the classification performance of different classifiers for the same feature. In addition, in the experiment, CNN was used to extract the depth features of three different datasets of KDDcup99, NSL-KDD and Gas Pipeline to illustrate the applicability of the CNN model.

Figure 7 shows the ROC curves of the three different datasets of KDDcup99, NSL-KDD and Gas Pipeline in C5.0, RF, SVM, C.4.5 and KNN.

It can be seen from Figure 7 that the classification AUC values of the C5.0 classifier in the three datasets KDDcup99, NSL-KDD and Gas Pipeline are 99.7%, 92.8% and 98.8%, respectively. In Figure 7(a), there are repeated records and large amount of data in KDDcup99 dataset, which results in the weak performance of KNN classifier. In Figure 7(b), because the NSL-KDD dataset removes redundant data and there are a certain number of unknown attacks, the classification performance of the KNN classifier is weaker than that of other classifiers. In Figure 7(c), the reason for the low AUC value of the SVM and KNN classifiers is that the synthesis minority oversampling methods solves the problem of unbalanced data classes in the Gas Pipeline dataset. The sample classes are combined with close neighbors to affect the SVM and KNN, The nearest neighbor synthesis of sample classes affects the classification performance of SVM and KNN.

Table 5 shows the two-class performance comparison between the method C5.0 and the four machine learning classifiers of RF, SVM, C4.5 and KNN under the three indicators of Acc, Precision and FAR.

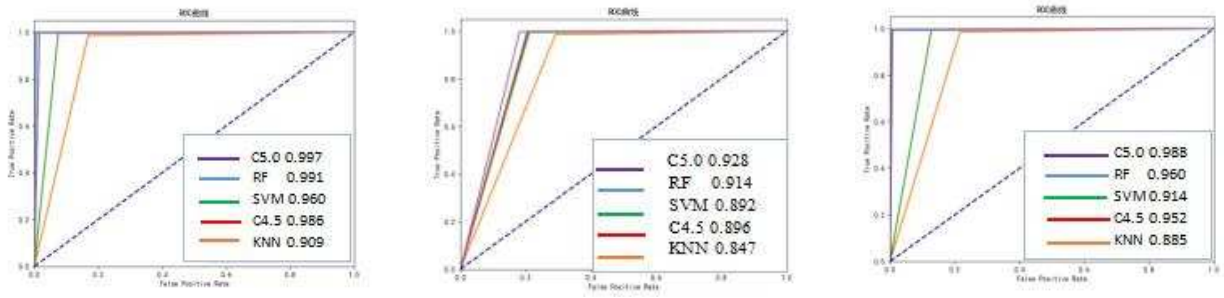
As can be seen from Table 5 that the classification Acc values of the C5.0 classifier used in this paper in the three data sets of KDDcup99, NSL-KDD and Gas Pipeline are 99.87%, 98.23% and 99.80%, respectively, compared with the performance of the other four classifiers optimal. KNN has the lowest classification performance on the KDDcup99 and NSL-KDD datasets and the highest false alarm rate. The SVM classifier has the lowest classification performance on the Gas Pipeline dataset, and the highest false alarm rate is still KNN. Therefore, the performance of the proposed method is the best.

5.4.3 Performance Comparison of CNN Features + C5.0 Multi-classification Experiments

Table 6 shows the detection results of the four different attack types and normal traffic in the KDDcup99 dataset in this paper, and compares them with the Acc of the SMOTE+ENN method [15] and the RTMAS method [2].

It can be seen from Table 6 that the proposed method has achieved 95.42%, 93.29%, 69.23%, and 86.90% for the attack test ACC of Dos, Probe, U2R, and R2L in the KDDcup99 dataset, respectively, and the classification Acc for Normal state reaches 98.69%, compared with the Acc and Normal Acc of the three attacks of Probe, U2R, and R21 obtained in the Ref. [15] are higher, while the proposed method has lower Acc for Dos detection. Compared with the Acc obtained in the Ref. [2], the proposed method has a higher Acc for Normal and Probe, U2R, and R2l attacks, but the detection accuracy rate for Dos attacks is still low, mainly because of the influence of the specific features selected by CNN on the detection accuracy of Dos attack. In some attacks, the similarity between U2R attacks and normal samples is as high as 100%, when the detection rate of normal behavior increases, the detection accuracy of U2R decreases compared to other attacks.

Table 7 shows the detection results of four different attack types and normal traffic in the NSL-KDD dataset by the proposed method, and compares them with the



(a) KDDcup99 (b) NSL-KDD (c) Gas Pipeline

Figure 7: Comparison of ROC curves of different datasets under five classifiers

Table 5: Comparison of the two classification performance of different classifiers with CNN features

Classifier	KDDcup99			NSL-KDD			Gas Pipeline		
	Acc(%)	Precision(%)	FAR(%)	Acc(%)	Precision(%)	FAR(%)	Acc(%)	Precision(%)	FAR(%)
C5.0	99.87	99.87	0.96	98.23	98.0	1.28	99.80	98.63	1.68
RF	98.17	97.69	1.43	96.82	95.96	1.36	97.40	96.67	2.16
SVM	95.62	94.05	2.02	93.14	92.89	2.05	92.53	92.08	3.65
C4.5	97.60	97.41	1.80	97.45	96.40	1.92	96.59	95.86	2.60
KNN	93.72	93.01	3.45	91.06	90.67	2.56	93.28	92.78	3.89

Table 6: The detection accuracy of five classifications based on KDDcup99 dataset

Method	Acc(%)				
	Normal	DoS	Probe	U2R	R2L
The proposed method	98.69	95.42	93.29	69.23	86.90
SMOTE+ENN [15]	98.68	97.23	84.38	55.0	42.02
RTMAS [2]	97.87	99.79	91.86	24.68	35.90

Acc of the AEML method [6] and the CNN method [11].

It can be seen from Table 7 that the proposed method has achieved 98.16%, 96.08%, 86.34%, and 90.18% in the attack test of Dos, Probe, U2R, and R2L in the NSL-KDD dataset, respectively, and the classification Acc for Normal reaches 97.03%. Compared with Acc in Ref. [6], the proposed method has a higher detection accuracy for DoS, Probe, U2R, R2L and Normal. Also compared with the Acc in Ref. [11], the detection accuracy of the proposed method for R2L attacks is 1.64% lower than that of the Ref. [11], and the detection accuracy of the other attacks is higher.

In order to further verify the performance of the proposed model, this paper uses the Gas Pipeline dataset to conduct a verification experiment. Table 8 shows the eight-category Acc detection performance of the proposed method on the Gas Pipeline dataset, and compares it with the detection accuracy of the GoogLeNet-LSTM method [4] and the CNN-BiLSTM method [24].

As can be seen from Table 8, the detection accuracy of the proposed method in NMRI, CMRI, MSCI MPCI, DoS and Normal is significantly higher than that in Ref. [4]. The detection accuracy of the GoogLeNet-LSTM method in Ref. [4] for both Recon and MFCI attacks is 100%, and the detection accuracy of the proposed method is lower than the Acc in Ref. [4]. Also compared with Ref. [24], the detection accuracy rate of the proposed method for NMRI, MSCI, MPCI, DoS, MFCI attacks is higher, while the detection accuracy rate for Normal, Recon and CMRI is slightly lower than that of the Ref. [24].

5.5 Performance Comparison with Existing Methods

In order to further highlight the performance of the proposed method, the performance of the proposed method is compared with the existing method [1, 2, 4, 9, 11, 13, 14, 19, 20, 23, 25], Table 9 and Table 10 are the results of binary classification and multiple classification performance comparison between the proposed method and the existing method on three different datasets: KDDcup99, NSL-KDD and Gas Pipeline.

It can be seen from Table 9 that the Acc value based on the proposed model in the KDDcup99 dataset is higher than AIDM-DL4JMLP [19] and SMOTE-RF [25]. The proposed method is higher than DL-AIDS [20] and CNN [11] in the two-class detection Acc on the NSL-KDD

Table 7: The detection accuracy of five classifications based on NSL-KDD dataset

Method	Acc(%)				
	Normal	DoS	Probe	U2R	R2L
The proposed method	97.03	98.16	96.08	86.34	90.18
AEML [6]	94.93	84.37	87.11	25	55.27
CNN [11]	82.40	92.19	64.32	64.52	91.82

Table 8: The detection accuracy of eight classifications based on Gas Pipeline dataset

Method	Acc(%)							
	Normal	Recon	NMRI	CMRI	MSCI	MPCI	MFCI	DoS
The proposed method	98.45	97.10	96.82	97.64	98.55	99.02	98.90	98.20
GoogLeNet-LSTM [4]	97.83	100	96.50	96.78	96.97	97.21	100	97.33
CNN-BiLSTM [24]	99.8	100	92.2	98.8	93.9	98.0	91.8	98.1

Table 9: Results of two-class classification performance comparison with existing methods

Method	Dataset	Acc(%)	FAR(%)
AIDM-DL4JMLP [19]	KDDcup99	97.9	N/A
SMOTE-RF [25]	KDDcup99	92.57	N/A
DL-AIDS [20]	NSL-KDD	77.99	0.4
CNN [11]	NSL-KDD	85.07	9.71
AEDL-ICS [1]	Gas Pipeline	95.86	N/A
CNN-ICS [9]	Gas Pipeline	99.46	N/A
The proposed method	KDDcup99	99.87	0.96
	NSL-KDD	98.23	1.28
	Gas Pipeline	99.80	1.68

dataset, but the FAR of CNN [11] is better than the proposed method. In the model detection task based on the Gas Pipeline industrial control network data set, the proposed model detection Acc is higher than AEDL-ICS [1] and CNN-ICS [9].

According to the data in Table 10, in the multi-classification task detection, the detection Acc of the proposed method in KDDcup99 dataset is higher than RTMAS-AIDS [2], and FAR lower than RTMAS-AIDS [2], but the detection accuracy and false positives rate of ANID-SEoKELM [14] are both better than that of the proposed method. For the five classification tasks of the NSL-KDD dataset, the detection accuracy of the proposed method is higher than ANID-CELM [23] and GA-GOGMM [13], but the false alarm rate of GA-GOGMM [13] is lower. In the eight-class detection task of the Gas Pipeline industrial control network dataset, the detection accuracy and false alarm rate of the proposed method are better than those of GoogLeNet-LSTM [4], and the detection accuracy of the CNN-ICS [9] model is higher than that of the proposed method. 0.34%.

Based on the above comparison and analysis, it can be seen that the five-class detection Acc on the NSL-KDD dataset is 98.54% higher than the two-class detection Acc 98.23%. The main reason is that when CNN is used to select the features of the NSL-KDD dataset, more features in the five categories are retained, which improves the detection accuracy of the five categories. In addition, compared with other models, the proposed method has higher detection accuracy on the three datasets of KDDcup99, NSL-KDD and Gas Pipeline, mainly because the proposed method uses a synthetic minority oversampling method to solve the problem of data imbalance in the dataset. In response to new attacks, the proposed method uses an online update strategy to ensure the adaptability of the proposed model, thereby improving the overall performance of the proposed model.

6 Conclusions and Future Work

An adaptive intrusion detection model based on CNN and C5.0 classifier is proposed, which improves the adaptability of the intrusion detection model in industrial control network to the dynamic changes of the network environment. The main work is reflected as follows: 1) The problem of data class imbalance is solved by using a few synthetic oversampling methods in industrial control network intrusion detection; 2) The effective data features are extracted by inputting the preprocessed data into the CNN model and adopting C5.0 classifier training and detection; 3) By introducing the adaptive online updating strategy based on frequent pattern mining, the updating of normal library and exception library is realized, which ensures the adaptability of the proposed model and realizes the detection of various attack behaviors. Experimental results show that the proposed method has good detection performance in terms of detection accu-

Table 10: Results of multi-classification performance comparison with existing methods

Method	Dataset	Acc(%)	FAR(%)
RTMAS-AIDS [2]	KDDcup99	95.86	2.13
ANID-SEoKELM [14]	KDDcup99	99.53	0.12
ANID-CELM [23]	NSL-KDD	84	3.03
GA-GOGMM [13]	NSL-KDD	96.52	1.43
GoogLeNet-LSTM [4]	Gas Pipeline	97.56	2.42
CNN-ICS [9]	Gas Pipeline	99.3	N/A
The proposed method	KDDcup99	98.18	1.43
	NSL-KDD	98.54	1.64
	Gas Pipeline	98.96	1.82

racy. The detection accuracy of KDDcup99, NSL-KDD and Gas Pipeline datasets reaches 98.18%, 98.54%, and 98.96%, respectively. At the same time, compared with the latest methods, whether it is two-class or multi-class, the detection accuracy of the proposed method is higher, and it can adapt to changes in the network environment.

The disadvantage is that the proposed method improves the detection accuracy while the false alarm rate is high. Further research will consider reducing the false alarm rate of the adaptive intrusion detection model and the long model training time.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61862041, 61363078), Scientific Research Program of Education Department of Shaanxi Province(No.19JK0040) and Science and Technology Project of Shaanxi Province (No.2020GY-041), The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] A. Al-Abassi, H. Karimipour, A. Dehghantanha, and R. M. Parizi, "An ensemble deep learning-based cyber-attack detection in industrial control system," *IEEE Access*, vol. 8, pp. 83965–83973, 2020.

[2] W. L. Al-Yaseen, Z. A. Othman, and M. Z. A. Nazri, "Real-time multi-agent system for an adaptive intrusion detection system," *Pattern Recognition Letters*, vol. 85, pp. 56–64, 2017.

[3] S. T. Bakhsh, S. Alghamdi, and R. A. Alsemmeari, "An adaptive intrusion detection and prevention system for internet of things," *International Journal of Distributed Sensor Networks*, vol. 15, no. 11, pp. 1–9, 2019.

[4] A. K. Chu, Y. X. Lai, and J. Liu, "Industrial control intrusion detection approach based on multiclassification googlenet-lstm model," *Security and Communication Networks*, vol. 2019, pp. 1–11, 2019.

[5] R. H. Dong, X. Y. Li, Q. Y. Zhang, and H. Yuan, "Network intrusion detection model based on multivariate correlation analysis-long short-time memory network," *IET Information Security*, vol. 14, no. 2, pp. 166–174, 2020.

[6] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An adaptive ensemble machine learning model for intrusion detection," *IEEE Access*, vol. 7, pp. 82512–82521, 2019.

[7] Y. Hu, A. Yang, H. Li, Y. Y. Sun, and L. M. Sun, "A survey of intrusion detection on industrial control systems," *International Journal of Distributed Sensor Networks*, vol. 14, no. 8, pp. 1–14, 2018.

[8] X. Kan, Y. X. Fan, Z. J. Fang, and L. Gao, "A novel iot network intrusion detection approach based on adaptive particle swarm optimization convolutional neural network," *Information Sciences*, vol. 568, pp. 147–162, 2021.

[9] Y. Lai, J. Zhang, and Z. Liu, "Industrial anomaly detection and attack classification method based on convolutional neural network," *Security and Communication Networks*, vol. 2019, no. 9, pp. 1–11, 2019.

[10] Y. Li, Y. Xu, and Z. Liu, "Robust detection for network intrusion of industrial iot based on multi-cnn fusion," *Measurement*, vol. 154, pp. 1–10, 2020.

[11] S. Z. Lin, Y. Shi, and Z. Xue, "Character-level intrusion detection based on convolutional neural networks," in *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Riode Janeiro, July 2018.

[12] J. Ling, Z. S. Zhu, Y. Lou, and H. Wang, "An intrusion detection method for industrial control systems based on bidirectional simple recurrent unit," *Computers and Electrical Engineering*, vol. 91, pp. 1–10, 2021.

[13] J. Liu, W. Zhang, and Z. Tang, "Adaptive intrusion detection via ga-gogmm-based pattern learning with fuzzy rough set-based attribute selection," *Expert Systems with Applications*, vol. 139, pp. 1–17, 2020.

[14] J. P. Liu, J. Z. He, and W. X. Zhang, "Anid-seokelm: Adaptive network intrusion detection based on selective ensemble of kernel elms with random features,"

Knowledge Based Systems, vol. 177, pp. 104–116, 2019.

[15] T. Lu, Y. P. Huang, W. Zhao, and J Zhang, "The metering automation system based intrusion detection using random forest classifier with smote+enn," in *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, pp. 370–374, Dalian, China, Oct 2019.

[16] Y. Luo, "Research on network security intrusion detection system based on machine learning," *International Journal of Network Security*, vol. 23, no. 3, pp. 490–495, 2021.

[17] B. Mahapatraa and S. Patnaik, "Self adaptive intrusion detection technique using data mining concept in an ad-hoc network," *Procedia Computer Science*, vol. 92, pp. 292–297, 2016.

[18] P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, "A detailed investigation and analysis of using machine learning techniques for intrusion detection," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 2, pp. 686–728, 2019.

[19] M. R. Mohamed, A. A. Nasr, I. F. Tarrad, and S. R. Abdulmageed, "Exploiting incremental classifiers for the training of an adaptive intrusion detection model," *International Journal of Network Security*, vol. 21, no. 2, pp. 1–15, 2019.

[20] D. Papamartzivanos, F. G. Marmol, G. Kambourakis, "Introducing deep learning self-adaptive misuse network intrusion detection systems," *IEEE Access*, vol. 7, pp. 13546–13560, 2019.

[21] H. C. Qu, Z. L. Qiu, X. M. Tang, M Xiang, and P Wang, "An adaptive intrusion detection method for wireless sensor networks," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 11, pp. 27–36, 2017.

[22] P. A. A. Resende and A. C. Drummond, "Adaptive anomaly-based intrusion detection system using genetic algorithm and profiling," *IEEE Communications Surveys and Tutorials*, vol. 1, no. 4, pp. 1–13, 2018.

[23] S. Roshan, Y. Miche, A. Akusok, and A. Lendasse, "Adaptive and online network intrusion detection system using clustering and extreme learning machines," *Journal of the Franklin Institute*, vol. 354, no. 4, pp. 1751–1779, 2018.

[24] L. Y. Shi, H. Q. Zhu, W. H. Liu, and J. Liu, "Industrial control system intrusion detection based on related information entropy and cnn-bilstm," *Journal of Computer Research and Development*, vol. 56, no. 11, pp. 2330–2338, 2019.

[25] X. P. Tan, S. J. Su, and Z. P. Huang, "Wireless sensor networks intrusion detection based on smote and the random forest algorithm," *Sensors*, vol. 19, no. 1, pp. 1–15, 2019.

[26] Y. Yang, H. Xu, L. Gao, Y. Yuan, K. McLaughlin, and S. Sezer, "Multidimensional intrusion detection system for iec 61850-based scada networks," *IEEE Transactions on Power Delivery*, vol. 32, no. 2, pp. 1068–1078, 2017.

[27] F. Yu, G. Li, H. Chen, and Y. Guo, "A vrf charge fault diagnosis method based on expert modification c5.0 decision tree," *International Journal of Refrigeration*, vol. 92, pp. 106–112, 2018.

[28] H. Zhang, L. Huang, C. Q. Wu, and Z Li, "An effective convolutional neural network based on smote and gaussian mixture model for intrusion detection in imbalanced dataset," *Computer Networks*, vol. 177, pp. 1–10, 2020.

Biography

Hao Wen-tao. received his master's degree in computer science and technology from Lanzhou University of Technology in 2019, is now a teacher at the Network Information Center of Xi'an Aeronautical University, a CCF member. His research interests include network and information security, industrial control network security, intrusion detection, and blockchain, etc.

Lu Ye. Lecturer, Doctoral student, working in the School of Computer Science, Baoji University of Arts and Sciences. The main research direction is network and information security, industrial control network and Internet of things security, blockchain, etc.

Dong Rui-hong. Researcher, worked at school of computer and communication in Lanzhou university of technology. His research interests include network and information security, information hiding and steganalysis analysis, computer network.

Shui Yong-li. received a bachelor's degree in management from Jilin University of Finance and Economics in 2018. Currently, she is studying for a master's degree in Lanzhou University of Technology. The main research directions are network and information security, industrial control network security, intrusion detection, etc.

Zhang Qiu-yu. Researcher/Ph.D. supervisor, graduated from Gansu University of Technology in 1986, and then worked at school of computer and communication in Lanzhou University of Technology. He is vice dean of Gansu manufacturing information engineering research center, a CCF senior member, a member of IEEE and ACM. His research interests include network and information security, information hiding and steganalysis, multimedia communication technology.

An Efficient and Secure Identity-based Conditional Privacy-Preserving Authentication Scheme in VANETs

Xianglong Wang[1], Qiuting Chen[1], Zhenwan Peng[2], and Yimin Wang[1]
(Corresponding author: Yimin Wang)
The School of Information and Computer, Anhui Agricultural University, Anhui, China[1]
130 Changjiangxilu, Hefei, Anhui, P.R. China
Email: ymw@ahau.edu.cn
The School of Biomedical Engineering, Anhui Medical University, China[2]
(Received Jan. 20, 2022; Revised and Accepted Apr. 28, 2022; First Online May 1, 2022)

Abstract

The existing conditional privacy-preserving identity-based schemes confront the high cost of pseudonym generation and key leakage in Vehicular ad-hoc networks (VANETs). We propose a new anonymous authentication scheme based on identity, aiming to address these issues. In this scheme, the pseudonym of the vehicle is generated by a roadside unit (RSU), reducing the computational pressure of trust authority (TA) or other pseudonym-generating entities. The complete private key of the message signature consists of partial private keys of TA, RSU, and OBU. If adversaries want to generate a legal message signature, they need a complete signature key. As a result, malicious vehicles in VANETs will be easily revoked as long as RSU stops providing pseudonyms and corresponding private keys. The analysis and performance evaluation of the proposed scheme indicate that the scheme has low revocation cost and high message verification and communication efficiency.

Keywords: Vehicular Ad-hoc Networks (VANETs); Conditional Privacy-preserving; Revocation

1 Introduction

With the development of modern science and technology, car ownership increased year by year, but this also caused more road congestion and traffic accident probability. These unpleasant events will affect the driver' s driving state. Therefore, safe and efficient management of road traffic in such cities is an urgent requirement. The rapid development of wireless communication technology (such as GMS, LTE, WiMAX and 5G etc.) provides convenience for intelligent transportation system (ITS), and the traffic generated by thousands of vehicles has been efficiently managed. Vehicular ad-hoc networks (VANETs) play an important role in ITS. VANETs supports two communication modes: vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) [7]. Through these modes with proper communication technologies, unnecessary accidents could be avoided according to certain information like weather conditions, vehicle location, traffic conditions and road defects [8].

Although having so many benefits, VANETs, like other networks, still meet many problems that are related to authentication and privacy-preserving because of the transparency [1, 14, 17, 22]. Therefore, more and more researchers are paying attention to and studying conditional privacy-preserving authentication schemes. The existing privacy-preserving authentication schemes in VANETs can be classified into three typical authentication schemes: public key infrastructure (PKI) -based, group signature-based and identity-based. ID-based is more efficient and reliable, and it is also one of the most important research directions. So the proposed conditional privacy-preserving authentication scheme is ID-based. ID-based scheme needs to generate a large number of pseudonyms to meet the requirements of anonymity. In some schemes [6,16,23], signatures are generated by trusted authority (TA) or private key generator (PKG), which greatly increase the burden of TA/PKG pseudonyms generation and management. To solve this problem, in He *et al.*'s schemes [11], the pseudonym of the vehicle is generated with the participation of the master key in the TPD. However, side-channel attacks cause sensitive information leakage in tamper-proof device (TPD) [24], and system master key leaks can also make VANETs unsafe. So Wang *et al.* [21] proposed a scheme that does not pre-install the master key in the TPD and generate pseudonyms by the vehicle itself. It effectively prevents the leakage of the master key caused by side-channel attacks, but OBU has limited computing power and may not be able to efficiently generate pseudonyms and message signature. In the scheme pro-

posed by Xiong *et al.* [23], the efficiency of message verification is very high, but the pseudonym is generated in batches by TA, which gives TA great computational pressure. And this scheme does not provide an efficient revocation method. Therefore, this research aims to propose an efficient scheme with conditional privacy-preserving based on Xiong *et al.* [23] and Wang *et al.* [21]. This scheme supports revocation, reducing TA's calculation pressure. The important contributions of this paper include the following:

- We propose a new scheme, which has higher security than existing schemes. Three parts consisted of private key of the signature: the system master key, RSU's private key and OBU's private key. Lacking any part of the key cannot generate a valid signature of message.
- Considering only elliptic curves will be used, our scheme has high verification efficiency and more adaptable to OBU which has limited calculation ability.
- This scheme is able to revoke malicious vehicles efficiently. When a malicious vehicle appears, it can be revoked as long as RSU stops updating its private key.

The composition of the paper is as follows. In Section 2, we introduce briefly the related work on conditional privacy-preserving schemes for VANETs. In Section 3, the background knowledge required for the system model of VANETs based on the proposed scheme is introduced in detail. In Section 4, we describe specifically the proposed scheme. In Section 5, we analyze the security of the proposed scheme. In Section 6, we conduct performance evaluation including validation and communication cost. Finally, we conclude the scheme in Section 7.

2 Related Work

In the introduction, there are existing problems such as communication security, vehicle anonymity and efficiency in VANETs. To improve the security and efficiency of VANETs, researchers have proposed a variety of conditional privacy-preserving authentication schemes for recent years. The existing conditional Privacy-Preserving schemes for VANETs can be divided into three types of authentication schemes: public key infrastructure (PKI) -based, group signature-based and identity-based.

In 2004, Hubaux *et al.* [13] pointed out security and privacy problems in vehicle communication for the first time and proposed a PKI-based scheme. In 2017, Azees M *et al.* [2] proposed a conditional tracking mechanism to trace malicious vehicles or RSUs. In 2021, EF Cahyadi *et al.* [4] proposed an improvement applying a Nonce in the final message. However, the PKI-based scheme requires huge communication overhead due to the storage and management of the certificate lists and the huge computation on the user side.

In 1991, the concept of group signature was proposed by Chaum and van Heyst [5]. In group signature, members of the group are anonymous and can verify the validity of received signature. In 2008, Hao *et al.* [10] proposed a distributed key management scheme which RSU distributes group private keys of a localized way. In 2009, Zhang *et al.* [25] proposed a distributed group authentication scheme, RSU maintains and manages vehicles within their communication range and include vehicles in temporary group. The schemes [10, 25] solve effectively the problems of vehicle privacy protection and the revocation of malicious vehicle in VANETs, but semi-trusted RSU may be attacked. Generally, the group-signature based schemes have the problems of the selection and credibility of group manager and the calculation in group signature.

In 2001, Rives *et al.* [18] proposed the concept of ring signature for the first time. Ring signature is special group signature, in which ring members equally rank and have no administrator. In 2018, Han *et al.* [9] proposed a dual protection scheme for VANETs through RSU auxiliary rings and security data communication. In 2020, Wang *et al.* [20] applied ring selection algorithm to VANETs and select ring members by ring selection algorithm. Obviously, the ring-based signature scheme has a higher level of privacy protection. However, tracking the real identity of malicious vehicles and revoking malicious vehicles are still difficult problems with ring signature-based schemes.

In 1984, Shamir [19] proposed identity-based signature and cryptosystem firstly. In 2013, Lee and Lai [15] proposed an authentication of the batch scheme based on bilinear pairing to enhance the security of VANETs. Horng *et al.* [12] proposed proposed an identity-based verification scheme with higher security and efficiency after correction. In 2015, Lo and Tsai [16] presented a new conditional privacy-preserving authentication scheme based on the elliptic curve cryptosystem to enhance scheme efficiency. In 2019, an efficient certificateless public key signature (CL-PKS) scheme was proposed by Ali *et al.* [1] based on bilinear pairing, they included blockchain to their CL-PKS scheme to improve the security of VANET. In 2022, EF Cahyadi *et al.* [3] summarized recent identity-based batch verification (IBV) schemes and proposed feasible improvements.

In 2020, Wang *et al.* [21] proposed a scheme that does not preinstall the master key of TPD to prevent side channel attacks. However, the limited computing power of OBU cannot efficiently generate pseudonyms and message signature in [21]. Xiong *et al.* [23] claimed that the scheme [15] can not satisfy secure against forgery or the non-repudiation property and guarantee vehicle privacy. Therefore, Xiong *et al.* [23] proposed a cheme aiming at the security flaw in [15]. TA can track the real identity of malicious vehicles. However, the scheme [23] cannot solve the problem of malicious vehicle revocation in VANETs, it does not have revocability. Therefore, we propose

Table 1: Overview table of the advantages of the proposed scheme over existing schemes

	SR-1	SR-2	SR-3	SR-4
Ali *et al.*'s scheme [1]	✗	✓	✗	✓
Horng *et al.*'s scheme [12]	✗	✗	✗	✓
Azees *et al.*'s scheme [2]	✗	✓	✗	✓
Lo *et al.*'s scheme [16]	✓	✗	✓	✗
Wang *et al.*'s scheme [21]	✓	✓	✗	✓
Xiong *et al.*'s scheme [23]	✓	✓	✓	✗
The proposed scheme	✓	✓	✓	✓

[1] SR-1, SR-2, SR-3, SR-4 represent four factors for evaluating the security and efficiency of the scheme, namely no pairing verification, defense against private key stolen attacks, high verification efficiency and revocation, respectively.
[2] ✓:The requirement is satisfied. ✗:The requirement is not satisfied or uninvolved.

an identity-based conditional privacy protection scheme based on Wang *et al.* [21] and Xiong *et al.* [23]. The comparison of some schemes with the proposed scheme are listed in Table 1.

3 Preliminarties

In this section, we describe the system model, security model and mathematical assumptions required to build the proposed scheme.

3.1 System Model

As shown in Figure 1, a complete VANETs consists of trust authority (TA), roadside unit (RSU) fixed on the roadside and on-board unit (OBU) installed on vehicles. The main functions of each entity in VANETs system are described as below.

TA. It is a generally trusted and authoritative entity. TA takes charge of the entire VANETs master key. When VANETs is attacked by malicious vehicles, TA can conduct identity tracking and identity revocation of malicious vehicles through tracking agency (TRA), and remove malicious vehicles from VANETs to ensure the communication security of legitimate vehicles.

RSU. It is a bridge entity that transmits information indirectly. RSU can communicate with OBU through wireless dedicated short-range communication (DSRC) protocol, and can also communicate with TA and application server (AS) through wired network. Therefore, RSU is a bridge between vehicles and TA in VANETs. In our scheme, it is considered malicious but not offensive.

OBU. The vehicle unit OBU is loaded on the vehicle, which contains the tamper-proof device (TPD) module. Information can be transmitted between vehicles and external entities through various external interfaces. Each vehicle broadcasts road traffic information to nearby vehicles every 100–300 ms, such as road congestion and driving state of surrounding vehicles. The communication process is based on DSRC protocol.

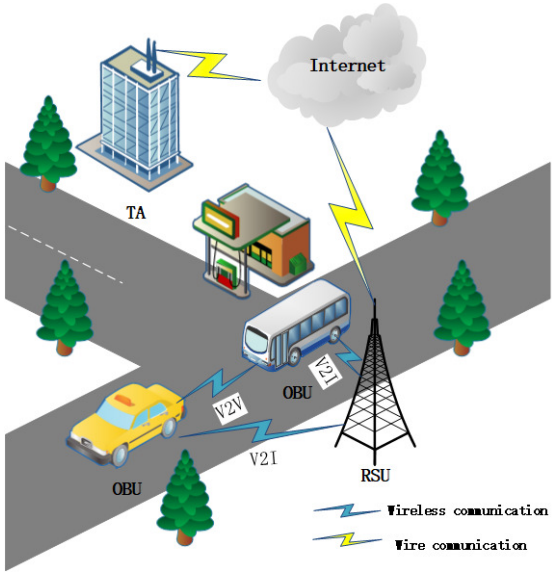


Figure 1: The system model

3.2 Security Model

A secure conditional privacy-preserving authentication scheme should meet the following security requirements.

Message Authentication and Integrity. All vehicle messages in VANETs should ensure that the message is not stolen and tampered by malicious third parties. When receiving the message, the recipient should verify whether the message is sent by the legitimate entity.

Anonymity. Vehicles and other vehicles in VANETs communication, the other vehicle cannot know the real original identity of the vehicle, that is, the receiving vehicle and the sending vehicle are anonymous in communication.

Unlink-ability. Unlink-ability refers that there is no correlation between different information sent by the same user, and the attacker cannot extract sensitive information from different information of the same user.

Traceability. TA can trace the true identity of a vehicle when a malicious vehicle sends malicious messages.

Revocation. If the malicious vehicle is tracked and confirmed, TA can revoke the malicious vehicle from VANETs.

3.3 Mathematic Assumption

First set a finite field $\mathbb{F}_p$, it has prime order p. Then set an elliptic curve defined by equation $y^2 = (x^3 + ax + b)$ mod p, where $a, b \in \mathbb{Z}_q^*$ and $(4a^3 + 27b^2)$ mod $p \neq 0$. An additive elliptic curve group $\mathbb{G}$ of order q is formed by defining $\mathbb{O}$ and some other points on the curve, where q is also a prime and P is the generator of $\mathbb{G}$.

Definition 1. *Elliptic curve discrete logarithm problem (ECDLP): There are two points $(P, W) \in \mathbb{G}$ are given. We consider that no probabilistic polynomial time (PPT) algorithm can calculate the random number $a \in \mathbb{Z}_q^*$ with an unnegligible probability, where a satisfies $W = a \cdot P$.*

Definition 2. *Computational Diffie-Hellman problem (CDHP): On the elliptic curve, some points $\{P, X = a \cdot P, W = b \cdot P\} \in \mathbb{G}$ are given, we consider that no PPT algorithm can calculate $a \cdot b \cdot P \in \mathbb{G}$ with an unnegligible probability, where $a, b \in \mathbb{Z}_q^*$.*

4 The Proposed Scheme

To meet the requirements of conditional privacy-preserve and high-level efficiency authentication in VANETs, we propose a new privacy-preserving scheme. In the proposed scheme, the master key is not preloaded to TPD and the pseudonym generation is executed by RSU. This scheme combines the master key, the private key of the RSU, the virtual ID of the vehicle and generates pseudonyms in the RSU. The scheme consists of six stages: (1) system initialization stage, (2) registration stage, (3) pseudonym and partial key generation stage, (4) key generation stage, (5) message signature stage, (6) message verification stage. Some definitions of notations are shown in Table 2.

Table 2: Notations and description used

Notation	Descriptions
s	The master key of the system
P_{pub}	The pubic key of the system
V_j	The j-th vehicle
RID_j	The real identity of V_j vehicle
VID_j	The vehicle V_j's token issued by TA
$PID_{j,i}$	The i-th pseudonym of the vehicle V_j
t_{r_k}	The k-th RSU's current private key
T_{r_k}	The k-th RSU's current public key
V_{sk_j}	The private key of vehicle V_j
V_{pk_j}	The pubic key of vehicle V_j
H_i	Secure Hash function
$E_{pk}(.)/D_{sk}(.)$	The encryption and the decryption of $Fhomo$
tt_i	Timestamp
$\|$	The message concatenation operation
$\oplus$	The exclusive-OR operation

4.1 System Initialization Stage

System initialization includes TA initialization and RSU initialization. TA is initialized by generating parameters, it selects randomly $s \in \mathbb{Z}_q^*$ and calculates $P_{pub} = s \cdot P$, in which P_{pub} and s are served as public key and master private key of the system, respectively. Then, TA selects two secure hash functions: $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$; $H_2 : \{0,1\} \times \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ and a homomorphic encryption $Fhomo$. Finally, TA transmits system parameters $\{P, P_{pub}, H_1, H_2, Fhomo\}$ to all RSUs and vehicles. RSU also requires initialization of parameters. The k-th RSU selects a random number $t_{r_k} \in \mathbb{Z}_q^*$ as its private key and calculates $T_{r_k} = t_{r_k} \cdot P$, then broadcasts T_{r_k} as its public key to all vehicles in the area.

4.2 Registration Stage

Vehicles must register offline to TA before they join VANETs. Vehicle V_j submits real identity RID_j to TA for validation(this identity must be legal in real life such as owner's identity card or license plate, as it is a necessary condition for tracking entity identity). If RID_j is valid, TA selects randomly a number $\alpha_{j,i} \in \mathbb{Z}_q^*$ as part of the vehicle's message signature key, it calculates $VID_j = RID_j \oplus \alpha_{j,i} \cdot P_{pub}$ as a virtual ID of the vehicle V_j in VANETs. Then TA selects randomly a number $V_{sk_j} \in \mathbb{Z}_q^*$, and compute $V_{pk_j} = V_{sk_j} \cdot P$ where V_{sk_j} is the private key of V_j and V_{pk_j} is the public key of V_j. Finally, parameter $pv_j = \{VID_j, SIG_s(VID_j), V_{pk_j}, V_{sk_j}\}$ is preloaded into TPD to generate pseudonyms and partial keys. TPD does not storage sensitive parameters in this step. The process of the registration stage is shown in Algorithm 1.

Algorithm 1 Vehicle Registration (Executed by TA)

Input: the system master key s, the vehicle V_j real identity RID_j.

Output:

1: Selects a random number $\alpha_{j,i}, V_{sk_j} \in \mathbb{Z}_q^*$
2: Computes $VID_j = RID_j \oplus \alpha_{j,i} \cdot P_{pub}$
3: Computes the signature $SIG_s(VID_j)$
4: Computes $V_{pk_j} = V_{sk_j} \cdot P$
5: **return** $pv_j = \{VID_j,\ SIG_s(VID_j),\ V_{pk_j},\ V_{sk_j}\}$

4.3 Pseudonym and Partial Key Generation Stage

When the vehicle V_j enters a new RSU area, the OBU will submits $\{VID_j,\ SIG_s(VID_j),\ V_{pk_j}\}$ to the RSU. When RSU receives the message, it will retransmit the message to TA for verifying the legitimacy of the vehicle. If the vehicle is legal, TA will return the tuple $\{\epsilon_{j,i},\ Q_j\}$ to RSU, where $\epsilon_{j,i} = E_{T_{r_k}}(s + \alpha_{j,i})$ and $Q_j = \alpha_{j,i} \cdot P$. Finally, the RSU calculates and returns the tuple $\{A_{j,i}, PID_{j,i}, E_{V_{pk_j}}(\delta_{j,i})\}$ $(i = 1, ..., n\ )$ to the vehicle, where $\delta_{j,i}$ is a partial signature key. Process as shown in Algorithm 2 to calculate parameters.

Algorithm 2 Generation of Pseudonym and Private Key (Executed by RSU)

Input: the ciphertext $\{\epsilon_{j,i}, Q_j\ \}$ $(i = 1, ..., n)$.

Output:

1: Selects a random number $k_{j,i} \in \mathbb{Z}_q^*$
2: Then computes

$$\begin{cases} B_{j,i} = D_{t_{r_k}}(\epsilon_{j,i}) + t_{r_k} = s + \alpha_{j,i} + t_{r_k} \\ PID_{j,i} = VID_j \oplus H(k_{j,i} \parallel P_{pub} \parallel Q_j) \\ h_{j,i} = H_1(PID_{j,i} \parallel P_{pub} \parallel T_{r_k} \parallel Q_j) \\ A_{j,i} = k_{j,i} \cdot P + h_{j,i} \cdot Q_j \\ \delta_{j,i} = k_{j,i} + h_{j,i} \cdot B_{j,i} \end{cases}$$

3: **return** $\{A_{j,i}, PID_{j,i}, E_{V_{pk_j}}(\delta_{j,i})\ \}$ $(\ i = 1, ..., n\ )$

4.4 Key Generation Stage

If a vehicle needs to communicate with another vehicle or RSU, OBU needs to sign a message and attach a timestamp to generate a message tuple.

The tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$ are calculated when the vehicle receives the tuple $\{A_{j,i}, PID_{j,i}, E_{V_{pk_j}}(\delta_{j,i})\ \}$ returned by the RSU, as illustrated in Algorithm 3.

Finally, the OBU broadcasts message tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$ to RSU and all vehicles in the area, and $\gamma_{j,i}$ is the signature of the message.

Algorithm 3 Signature Generation (Executed by OBU)

Input: the ciphertext $\{A_{j,i}, PID_{j,i}, E_{V_{pk_j}}(\delta_{j,i})\}$.

Output:

1: Computes $h_{j,i} = H_1(PID_{j,i} \parallel P_{pub} \parallel T_{r_k} \parallel Q_j)$
2: Computes $h_{j,i}^{'} = H_2(PID_{j,i} \parallel M \parallel T_{r_k} \parallel tt_i)$
3: Computes $\delta_{j,i} = D_{V_{sk_j}}(\delta_{j,i}) = k_{j,i} + h_{j,i} \cdot B_{j,i}$
4: Computes $\gamma_{j,i} = \delta_{j,i} + h_{j,i}^{'} \cdot V_{sk_j}$
5: **return** $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$

4.5 Message Verification Stage

4.5.1 Single Verification

The message tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$ can be verified by the RSU or all vehicles in the area. At first, the recipient will check whether the timestamp tt_i is refreshed, if not, the message will be rejected, else the following equation will continue to be verified:

$$\gamma_{j,i} \cdot P == A_{j,i} + h_{j,i} \cdot (P_{pub} + T_{r_k}) + h_{j,i}^{'} \cdot V_{pk_j} \quad (1)$$

The recipient will trusts the message if Equation (1) is satisfied, or rejects the message if not.

If the message tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$ is not tampered in the transmission process, it will satisfy the equation(1). Since $\gamma_{j,i} = \delta_{j,i} + h_{j,i}^{'} \cdot V_{sk_j}$, $\delta_{j,i} = k_{j,i} + h_{j,i} \cdot B_{j,i}$ and $B_{j,i} = s + \alpha_{j,i} + t_{r_k}$, where $h_{j,i} = H_1(PID_{j,i} \parallel P_{pub} \parallel T_{r_k} \parallel Q_j)$, and $h_{j,i}^{'} = H_2(PID_{j,i} \parallel M \parallel T_{r_k} \parallel tt_i)$, so we have the following:

$$\begin{aligned} \gamma_{j,i} \cdot P &= (k_{j,i} + h_{j,i} \cdot B_{j,i} + h_{j,i}^{'} \cdot V_{sk_j}) \cdot P \\ &= k_{j,i} \cdot P + h_{j,i} \cdot (s + \alpha_{j,i} + t_{r_k}) \cdot P + h_{j,i}^{'} \cdot V_{sk_j} \cdot P \\ &= A_{j,i} + h_{j,i} \cdot (P_{pub} + T_{r_k}) + h_{j,i}^{'} \cdot V_{pk_j} \end{aligned}$$

Therefore, the scheme can correctly validate single messages. The process of message verification such as Algorithm 4.

4.5.2 Batch Verification

This scheme also supports batch verification of multiple messages received. When the recipient receives multiple messages, the recipient can verify whether Equation (2) satisfies.

$$\begin{aligned} \left(\sum_{j,i=0}^{n} (d_{j,i} \cdot \gamma_{j,i}) \right) \cdot P = & \sum_{j,i=0}^{n} d_{j,i} \cdot A_{j,i} \\ & + \left(\sum_{j,i=0}^{n} (d_{j,i} \cdot h_{j,i}) \right) \cdot (P_{pub} + T_{r_k}) \\ & + \sum_{j,i=0}^{n} \left((d_{j,i} \cdot h_{j,i}^{'}) \cdot V_{pk_j} \right) \end{aligned} \quad (2)$$

In the equation, $d_{1,i}, d_{2,i}, ..., d_{n,i} \in [1, 2^t]$, where t is a small integer.

Algorithm 4 Message Verification (Executed by Vehicle or RSU)

Input: the message tuple$\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$.

Output:

1: Checks whether the timestamp tt_i is refreshed, if not, rejects
2: **if** tt_i is fresh **then**
3: Computes $h_{j,i} = H_1(PID_{j,i} \parallel P_{pub} \parallel T_{r_k} \parallel Q_j)$
4: Computes $h'_{j,i} = H_2(PID_{j,i} \parallel M \parallel T_{r_k} \parallel tt_i)$
5: **if** $\gamma_{j,i} \cdot P == A_{j,i} + h_{j,i} \cdot (P_{pub} + T_{r_k}) + h'_{j,i} \cdot V_{pk_j}$ **then**
6: **return** true
7: **else**
8: **return** false
9: **end if**
10: **else**
11: **return** false
12: **end if**

The proof process is as follows :

$$\left(\sum_{j,i=0}^{n}(d_{j,i}\cdot\gamma_{j,i})\right)\cdot P$$
$$=\sum_{j,i=0}^{n} d_{j,i}\cdot(k_{j,i}+h_{j,i}\cdot B_{j,i}+h'_{j,i}\cdot V_{sk_j})\cdot P$$
$$=\sum_{j,i=0}^{n} d_{j,i}\cdot A_{j,i}+\left(\sum_{j,i=0}^{n}(d_{j,i}\cdot h_{j,i})\right)\cdot(P_{pub}+T_{r_k})$$
$$+\sum_{j,i=0}^{n}\left((d_{j,i}\cdot h'_{j,i})\cdot V_{pk_j}\right)$$

Therefore, the scheme can validate multiple messages correctly.

5 Scheme Analysis

In this section, we will analyze the security and privacy of our scheme.

5.1 Message Integrity

This scheme divides the key generation of message signature into three parts: the system master key, RSU's private key and OBU's private key. When missing any part of the key, the message signature cannot be generated. In addition, as long as ECDLP is difficult to be solved, the attacker cannot forge a vaild message signature. Therefore, if the signature and the message tuple satisfy the equation $\gamma_{j,i} \cdot P == A_{j,i} + h_{j,i} \cdot (P_{pub} + T_{r_k}) + h'_{j,i} \cdot V_{pk_j}$, authentication and integrity of the message can be guaranteed according to the above verification process.

5.2 Anonymity and Unlink-ability

In the process of generating pseudonyms, $PID_{j,i} = VID_j \oplus H(k_{j,i} \parallel P_{pub} \parallel Q_j)$, where $k_{j,i}$ is a number selected randomly by RSU without any valuable information, so the scheme can meet the requirements of anonymity. Each vehicle's message is sent under a different pseudonym. These pseudonyms that are randomly generated on RSU with no correlation, so the scheme can meet the requirements of Unlink-ability.

5.3 Traceability

The message tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$ that sent by the vehicle includes the pseudonym $PID_{j,i}$, where $PID_{j,i} = VID_j \oplus H(k_{j,i} \parallel P_{pub} \parallel Q_j)$, so TA can calculate

$$VID_j = PID_{j,i} \oplus H(k_{j,i} \parallel P_{pub} \parallel Q_j)$$
$$RID_j = VID_j \oplus \alpha_{j,i} \cdot P_{pub}$$

to get the real identity RID_j of the vehicle.

5.4 Revocation

When the real identity of the malicious vehicle is confirmed, it will be added to the revocation list, and TA will notify the RSU in the area where the malicious vehicle is located. RSU will update its private key t'_{r_k} and public key T'_{r_k} after receiving revocation instructions that sent by TA. RSU retransmits partial message signature key tuple $\{A'_{j,i}, PID'_{j,i}, E_{V_{pk_j}}(\delta'_{j,i})\}(i = 1, ..., n)$ to normal legitimate vehicles. However, the parameters of malicious vehicles are not updated, so the above proof process is not satisfied, namely $\gamma_{j,i} \cdot P \neq A_{j,i} + h_{j,i} \cdot (Pub + T'_{r_k}) + h'_{j,i} \cdot V_{pk_j}$, since T_{r_k} is obviously not equal to T'_{r_k}. Therefore, RSU and other vehicles no longer trust messages taht sent by malicious vehicles. So the scheme supports the revocation of malicious vehicles.

5.5 Resist Multiple Types of Attacks

In this subsection, we will demonstrate and analyze the ability of the scheme to resist five common attacks.

Simulating Attacks. Assume that an attacker can forge and generate a valid message tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$. This means that the attacker can forge the valid signature of vehicle V_j. We have already analyzed the reliability and integrity of the message of the scheme, that is, the attacker cannot forge a valid signature, because the forgery is impossible unless three partial keys are obtained at the same time. The probability of forging legitimate message signatures can be ignored.

Tampering Attacks. Suppose an attacker can forge and generate message tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$ of vehicle V_j. It means that the attacker can forge the valid signature

Table 3: Operations and description used

Operations	Descriptions	Time(ms)
t_{bp}	The execution time of a bilinear pairing operation	4.211
t_{bpm}	The execution time of a scalar multiplication operation related to the bilinear pairing	1.709
t_{bpa}	The execution time of a point addition operation related to the bilinear pairing	0.0071
t_{em}	The execution time of a scalar multiplication operation	0.442
t_{esm}	The execution time of a small scale multiplication operation	0.0138
t_{mtp}	The time to perform a MapToPoint operation	4.406

of vehicle V_j, but the operation process of message signature ensures the uniqueness of the message. This is almost impossible without solving the ECDLP.

Repeat Attacks. When the recipient receives the message, at first it will check whether the timestamp tt_i is refreshed. Repeated message tuple will be rejected by the recipient. Therefore, the scheme can resist repeated attacks.

Man-in-the-middle Attacks. In the above analysis, all messages must be signed, and the message signatures cannot be forged without obtaining the private key. Therefore, the proposed scheme can resist man-in-the-middle attack.

Private Key Stolen Attacks. In the scheme, the signature of the message requires completed system private key s, RSU's private key t_{r_k} and OBU's private key V_{sk_j}. Even if the system master key s or the vehicle private key V_{sk_j} are leaked to the adversary under a side-channel attack, it is still unable to generate a valid message signature. Therefore, the scheme can resist private key stolen attacks.

6 Performance Evaluation

In this section, we will evaluate the performance of the scheme, which includes verification and communication costs. In addition, we will compare the scheme with other existing schemes in VANETs. We set the security level to 80 bits and use an elliptic curve additive group $\mathbb{G}$, which means p and q are primes of two 160 bits. Here we use a bilinear pairing: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ to ensure that the security level is 80 bits, where $\mathbb{G}_1$ is the additive group on the elliptic curve, and the embedding degree is 2. We ignore the time required for general hash operation, XOR operation and general multiplication. In the scheme, the vehicle pseudonym is generated by RSU that has super computing power. Therefore, we do not consider to compare the time of signature generation in comparison. We adopt the experiment and evaluation method in [25]. According to the experiment in [25], we show that the execution time and description of the main encryption operations are listed in Table 3.

6.1 Verification Cost

The cryptographic operations in the schemes of Ali *et al.* [1], Azees *et al.* [2] and Horng *et al.* [12] are based on bilinear pairing, the scalar multiplications are performed on elliptic curves that is related to bilinear pairing. The cryptographic operations in the schemes of Wang *et al.* [21], Lo *et al.* [16], Xiong *et al.* [23] and the proposed scheme, the scalar multiplication is performed on a given elliptic curve. We will analyze the execution time of one message single verification and multiple message batch verification in detail for the above four schemes.

For the single verification of Ali *et al.* [1], the vehicle needs to execute one bilinear pairing operation, one scalar multiplication operation and one point addition operation that are related to bilinear pairing, therefore, the time that required to verify the single message is $1t_{bp}+1t_{bpm}+1t_{bpa} \approx 5.9271$ ms; for the batch verification of multiple messages, the verifier needs to execute one bilinear pairing operation, n scalar multiplication operations and n point additions operations that are related to bilinear pairing, therefore, the time that required to verify n messages is $t_{bp}+nt_{bpm}+nt_{bpa} \approx 1.7161n+4.211$ ms. Similarly, in the scheme of Horng *et al.* [12], the time that required to verify the single message is $2t_{bp}+2t_{bpm}+1t_{mtp} \approx 16.246$ ms, the time that required to verify n messages is $2t_{bp}+2nt_{bpm}+nt_{mtp} \approx 7.824n+8.422$ ms. In the scheme of Azees *et al.* [2], the time that required to verify the single message is $2t_{bp}+5t_{bpm}+2t_{bpa} \approx 16.9812$ ms, the time that required to verify n messages is $(n+1)t_{bp}+5nt_{bpm}+2nt_{bpa} \approx 12.7702n+4.211$ ms.

For the single verification of proposed scheme, the vehicle needs to execute three scalar multiplication operations and one small scale multiplication operation, therefore, the time that required to verify the single message is $3t_{em}+1t_{esm} \approx 1.3398$ms; for the batch verification of multiple messages, the verifier needs to execute $(n+2)$ scalar multiplication operations and n small scale multiplication operations, therefore, the time that required to verify n messages is $(n+2)t_{em}+nt_{esm} \approx 0.4558n+0.884$ ms. Similarly, in the scheme of Wang *et al.* [21], the time that required to verify the single message is $4t_{em} \approx$

Table 4: Comparison of verification cost

Schemes	Single verification(ms)	Batch verification (ms)
Ali *et al.*'s scheme [1]	$1t_{bp} + 1t_{bpm} + 1t_{bpa} \approx 5.9271$	$t_{bp} + nt_{bpm} + nt_{bpa} \approx 1.7161n + 4.211$
Horng *et al.*'s scheme [12]	$2t_{bp} + 2t_{bpm} + 1t_{mtp} \approx 16.246$	$2t_{bp} + 2nt_{bpm} + nt_{mtp} \approx 7.824n + 8.422$
Azees *et al.*'s scheme [2]	$2t_{bp} + 5t_{bpm} + 2t_{bpa} \approx 16.9812$	$(n+1)t_{bp} + 5nt_{bpm} + 2nt_{bpa} \approx 12.7702n + 4.211$
Lo *et al.*'s scheme [16]	$3t_{em} \approx 1.326$	$(n+2)t_{em} + 2nt_{esm} \approx 0.4696n + 0.884$
Wang *et al.*'s scheme [21]	$4t_{em} \approx 1.768$	$(2n+3)t_{em} + 2nt_{esm} \approx 0.9116n + 1.326$
Xiong *et al.*'s scheme [23]	$3t_{em} + t_{esm} \approx 1.3398$	$(n+2)t_{em} + nt_{esm} \approx 0.4558n + 0.884$
The proposed scheme	$3t_{em} + t_{esm} \approx 1.3398$	$(n+2)t_{em} + nt_{esm} \approx 0.4558n + 0.884$

1.768 ms, the time that required to verify n messages is $(2n+3)t_{em}+2nt_{esm} \approx 0.9116n+1.326$ ms. In the scheme of Lo *et al.* [16], the time that required to verify the single message is $3t_{em} \approx 1.326$ ms, the time that required to verify n messages is $(n+2)t_{em} + 2nt_{esm} \approx 0.4696n + 0.884$ ms. In the scheme of Xiong *et al.* [23], the time that required to verify the single message is $3t_{em} + t_{esm} \approx 1.3398$ ms, the time that required to verify n messages is $(n+2)t_{em} + nt_{esm} \approx 0.4558n + 0.884$ ms.

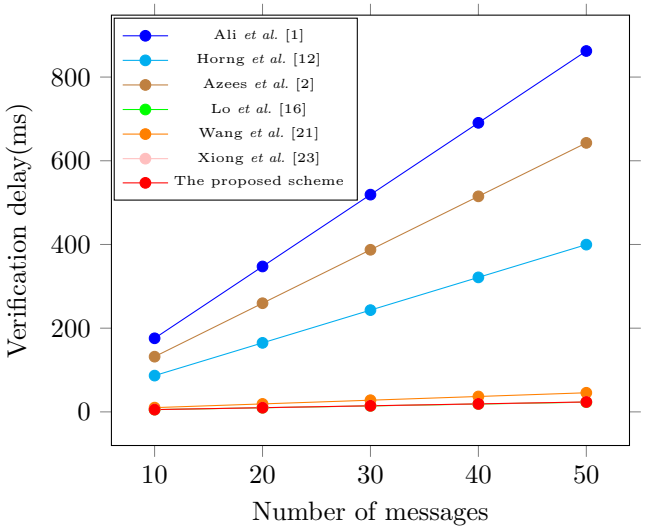


Figure 2: Verification operation cost of multiple messages

The calculation cost comparison of all schemes is listed in Table 4. Figure 2 shows a comparison of the verification cost of the scheme. According to Table 4 and Figure 2, the proposed scheme, the schemes of Lo *et al.* [16], Wang *et al.* [21] and Xiong *et al.* [23] have higher certification efficiency than the schemes of Horng *et al.* [12], Azees *et al.* [2] and Ali *et al.* [1], since these schemes use elliptic curve encryption instead of bilinear pairing. Compared with other schemes, the proposed scheme does not preload the master key of the system into TPD, and generate pseudonyms by RSU. Therefore, the proposed scheme has higher security and pseudonym generation efficiency.

6.2 Communications Cost

In this subsection, we will analyze the other communication costs of the proposed scheme, which includes the communication costs in addition to the message itself, such as signature, pseudonym, certificate and so on. Communication costs for five schemes are listed in Table 5.

Table 5: Comparison of communication cost

Schemes	a message(bytes)	n message (bytes)
Ali *et al.* [1]	536	536n
Horng *et al.* [12]	388	388n
Azees *et al.* [2]	848	848n
Lo *et al.* [16]	188	188n
Wang *et al.* [21]	124	124n
Xiong *et al.* [23]	128	128n
The proposed scheme	124	124n

In the scheme of Ali *et al.* [1], the vehicle broadcasts $\{AID_{i,1}, AID_{i,2}, X_i, Y_i, \theta, t_i\}$ to the recipient, where $AID_{i,1}, X_i, Y_i, \theta \in \mathbb{G}_1, AID_{i,2} \in \mathbb{Z}_q^*$ and t_i is the time stamp. Therefore, the communication cost is $4 * 128 + 20 + 4 = 536$ bytes. In Horng *et al.*'s scheme [12], the vehicle broadcasts $\{PID_i^1, PID_i^2, \sigma\}$ to the recipient, where $PID_i^1, PID_i^2, \sigma \in \mathbb{G}_1$, thus the communication cost is $3 * 128 + 4 = 388$ bytes. In Azees *et al.*'s scheme [2], the vehicle broadcasts its signature messages $\{sig \parallel Y_k \parallel Cert_k\}$ to the verifier, where $Cert_k = \{Y_k \parallel E_i \parallel DID_{ui} \parallel \gamma_u \parallel \gamma_v \parallel c \parallel \lambda \parallel \sigma_1 \parallel \sigma_2\}$, $\{sig, E_i, DID_{ui}, \gamma_u, \gamma_v, Y_k\} \in \mathbb{G}_1, \{\lambda, \sigma_1, \sigma_2\} \in \mathbb{Z}_q^*$, thus the communication cost is $6 * 128 + 4 * 20 = 848$ bytes. In Lo *et al.*'s scheme [16], the vehicle broadcasts $\{PID_i = (PID_{i,1}, PID_{i,2}, t_i), tt_i, \delta = (K_i, R_i, V_i)\}$ to the recipient, where $PID_{i,1}, K_i, R_i, V_i \in \mathbb{G}$, $PID_{i,2} \in \mathbb{Z}_q^*$ and t_i, tt_i are timestamps, thus the communication cost is $4*40+20+4*2 = 188$ bytes. In Wang *et al.*'s scheme [21], the vehicle broadcasts $\{PID_{i,j} = (PID1_{i,j}, PID2_{i,j}), U_{i,j}, V_{i,j}, tt_i\}$ to the recipient, where $PID2_{i,j}, U_{i,j} \in \mathbb{G}$, $PID1_{i,2}, V_{i,j} \in \mathbb{Z}_q^*$ and tt_i is timestamp, thus the communication cost is $2*40+2*20+4 = 124$ bytes. In Xiong *et al.*'s scheme [23],

the vehicle broadcasts $\{A_{j,i}, PID_{j,i}, S_{pub_j}, T_{j,i}, \beta_{j,i}, t_{j,i}\}$ to the recipient, where $A_{j,i}, S_{pub_j} \in \mathbb{G}$, $PID_{j,i}, \beta_{j,i} \in \mathbb{Z}_q^*$ and $T_{j,i}$, tt_i are timestamps, thus the communication cost is $2 * 40 + 2 * 20 + 4 * 2 = 128$ bytes. In the proposed scheme, the vehicle broadcasts message tuple $\{A_{j,i}, M, PID_{j,i}, V_{pk_j}, \gamma_{j,i}, tt_i\}$ to the surrounding receiving unit, where $A_{j,i}, V_{pk_j} \in \mathbb{G}$, $PID_{j,i}, \gamma_{j,i} \in \mathbb{Z}_q^*$ and tt_i is time stamp. Therefore the communication cost is $2 * 40 + 2 * 20 + 4 = 124$ bytes. The communication costs of the five schemes are shown in Figure 3. According to the Figure 3, the proposed scheme has a very low communication cost compared with other schemes.

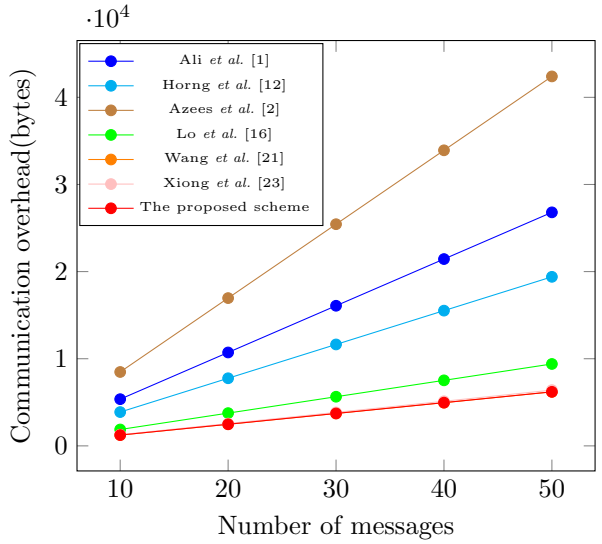


Figure 3: Communication cost of multiple messages

7 Conclusion

In this study, a secure and efficient conditional privacy-preservation scheme that based on identity for V2V and V2I communication in VANETs has been proposed. Since the signature key of the vehicle message is generated by the private key of TA, RSU and vehicle itself, the message will not be signed if any part of the private key is missing, so this scheme can stop attacker forging the message. In addition, the pseudonym is generated by RSU, which reduces the burden of TA calculation and pseudonym management. It also means that malicious vehicles can be effectively revoked from VANETs as long as the RSU stops providing pseudonyms and corresponding private keys. Performance evaluation results reveal that the scheme has higher verification efficiency and lower communication cost.

Acknowledgments

This research was supported by Anhui Provincial Natural Science Foundation (2108085QF274) and Talent Foundation of agricultural university (rc482005).

References

[1] I. Ali, M. Gervais, E. Ahene, and F. Li, "A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in vanets," *Journal of Systems Architecture*, vol. 99, p. 101636, 2019.

[2] M. Azees, P. Vijayakumar, and L. J. Deboarh, "Eaap: Efficient anonymous authentication with conditional privacy-preserving scheme for vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 9, pp. 2467–2476, 2017.

[3] E. F. Cahyadi, C. Damarjati, and M. S. Hwang, "Research on identity-based batch verification schemes for security and privacy in vanets," *Journal of Electronic Science and Technology*, vol. 20, no. 3, pp. 1–19, 2022.

[4] E. F. Cahyadi and M. S. Hwang, "An improved efficient anonymous authentication with conditional privacy-preserving scheme for vanets," *Plos one*, vol. 16, no. 9, p. e0257044, 2021.

[5] D. Chaum and E. Van Heyst, "Group signatures," in *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer, 1991, pp. 257–265.

[6] J. Cui, X. Zhang, H. Zhong, J. Zhang, and L. Liu, "Extensible conditional privacy protection authentication scheme for secure vehicular networks in a multi-cloud environment," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1654–1667, 2019.

[7] M. Gonzalez-Martín, M. Sepulcre, R. Molina-Masegosa, and J. Gozalvez, "Analytical models of the performance of c-v2x mode 4 vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1155–1166, 2018.

[8] M. M. Hamdi, L. Audah, S. A. Rashid, and M. Al Shareeda, "Techniques of early incident detection and traffic monitoring centre in vanets: A review." *Journal of Communications*, vol. 15, no. 12, pp. 896–904, 2020.

[9] Y. Han, N. N. Xue, B. Y. Wang, Q. Zhang, C. L. Liu, and W. S. Zhang, "Improved dual-protected ring signature for security and privacy of vehicular communications in vehicular ad-hoc networks," *IEEE Access*, vol. 6, pp. 20 209–20 220, 2018.

[10] Y. Hao, Y. Cheng, and K. Ren, "Distributed key management with protection against rsu compromise in group signature based vanets," in *IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference*. IEEE, pp. 1–5, 2008.

[11] D. He, S. Zeadally, B. Xu, and X. Huang, "An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2681–2691, 2015.

[12] S. J. Horng, S. F. Tzeng, Y. Pan, P. Fan, X. Wang, T. Li, and M. K. Khan, "b-specs+: Batch verification for secure pseudonymous authentication in vanet," *IEEE transactions on Information Forensics and Security*, vol. 8, no. 11, pp. 1860–1875, 2013.

[13] J. P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security & Privacy*, vol. 2, no. 3, pp. 49–55, 2004.

[14] Q. Jiang, J. Ma, G. Li, and X. Li, "Improvement of robust smart-card-based password authentication scheme," *International Journal of Communication Systems*, vol. 28, no. 2, pp. 383–393, 2015.

[15] C. C. Lee and Y. M. Lai, "Toward a secure batch verification with group testing for vanet," *Wireless networks*, vol. 19, no. 6, pp. 1441–1449, 2013.

[16] N. W. Lo and J. L. Tsai, "An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 5, pp. 1319–1328, 2015.

[17] B. Palaniswamy, S. Camtepe, E. Foo, and J. Pieprzyk, "An efficient authentication scheme for intra-vehicular controller area network," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3107–3122, 2020.

[18] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, pp. 552–565, 2001.

[19] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the theory and application of cryptographic techniques*. Springer, pp. 47–53, 1984.

[20] L. Wang, X. Lin, L. Qu, and C. Ma, "Ring selection for ring signature-based privacy protection in vanets," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, pp. 1–6, 2020.

[21] Y. Wang, H. Zhong, Y. Xu, J. Cui, and G. Wu, "Enhanced security identity-based privacy-preserving authentication scheme supporting revocation for vanets," *IEEE Systems Journal*, vol. 14, no. 4, pp. 5373–5383, 2020.

[22] L. Wu, Y. Xia, Z. Wang, and H. Wang, "Be stable and fair: Robust data scheduling for vehicular networks," *IEEE Access*, vol. 6, pp. 32 839–32 849, 2018.

[23] W. Xiong, R. Wang, Y. Wang, F. Zhou, and X. Luo, "CPPA-D: Efficient conditional privacy-preserving authentication scheme with double-insurance in vanets," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3456–3468, 2021.

[24] T. Zaidi and S. Faisal, "An overview: Various attacks in vanet," in *2018 4th International Conference on Computing Communication and Automation (ICCCA)*. IEEE, pp. 1–6, 2018.

[25] L. Zhang, Q. Wu, A. Solanas, and J. Domingo-Ferrer, "A scalable robust authentication protocol for secure vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1606–1617, 2009.

Biography

Xianglong Wang is a student at the School of Information and Computer Science, Anhui Agricultural University. His research interest is privacy security protection in wireless communication.

Qiuting Chen is a student at the School of Information and Computer Science, Anhui Agricultural University. Her research interests is network information security.

Zhenwan Peng is now a Lecturer in the School of Biomedical Engineering, Anhui Medical University. He received the Ph.D. degree from Anhui University of China, in 2018. His research interest includes classical and quantum cryptography, in particular, secure multiparty computations.

Yimin Wang is now an Associate Professor in the School of Computer Science and Technology, Anhui Agriculture University. He received PhD degree in Anhui University of China. His research interests include security and privacy for wireless networks, cloud computing, big data, etc..

A Technical Review on Network Security Situation Awareness

Wen Xi[1], Wei Wu[2], and Cheng-Ying Yang[3]
(Corresponding author: Cheng-Ying Yang)
Big Data Academy, Fuzhou University of International Studies and Trade, Fuzhou, Fujian, China[1]
Wuhan Digital Engineering Institute, Wuhan, Hubei, China[2]
Department of Computer Science, University of Taipei[3]
Email: cyang@utaipei.edu.tw
(Received Dec. 21, 2021; Revised and Accepted May 7, 2022; First Online May 8, 2022)

Abstract

Network security situation awareness is one of the most concerning research in network security. According to the progressive relationship of the subject, there are three stages, situation awareness, situation evaluation, and situation prediction needed to be implemented. This work emphasizes the technologies applied towards the model establishment, the estimation methods, and prediction schemes. Among these stages, the situation prediction is the most difficult but the most valuable. With the accurate prediction of network security status in advance, it can fundamentally improve network security performance. Therefore, the prediction of network security situations has received continuous attention. Besides, the introduction of the application to the industrial and Internet of Things is provided. This paper aims to summarize and sort out the system model, the leading technologies, and the application fields of network security situation awareness to provide a reference for relevant researchers in the security field.

Keywords: Estimation; Network Security; Prediction; Situation Awareness

1 Introduction

Network Security is an important issue in the growing up and extending networking environment. Although a great deal of researches and protection methods have been proposed, they usually could solve the isolated and independent problems. There are still problems in the network security, without the systematical solution existing, if those proposed solutions could not be integrated effectively. Network Security Situation Awareness (NSSA) [4] is a concept to integrate all available information and evaluates the security situation in the network. It could provide some security analyses to minimize the risks and to reduce the losses because of the unsafe factors. Upon the respects of monitoring capabilities and emergency response, NSSA could benefit and contribute for the network security [42].

To prevent the attackers and the network instruction, some technologies and the security products are employed, such as firewall. Firewall technology belongs to an access control scheme implemented in the network. The major purpose is to prevent a desired network from the attackers with a permission to control the entering and existing of the network. According to the different schemes adopted, the technologies could be categorized as packet filtering, network address translation, proxy and monitoring types [7].

1) Packet Filter:
The products of packet filtering are the primary devices of firewalls. The advantage with this technology is simple and practical. The cost of implementation is low. In the case of a relatively simple application, the system security could be guaranteed to a certain extent based on a lower cost.

2) Network Address Translation - NAT:
Internet Address Translation is a standard technology to convert the IP address to a temporary, external and registered IP address. The process of network address translation is transparent to the user. It does not require the user to set up. Also, the user could operate normally.

3) Proxy:
The firewall is also called a proxy server. The proxy server is located between the client and the server. It could completely block the data exchange. The advantage of proxy server is with a high level of security. It could detect and scan the application layer. Also, it is very effective against the intrusions and viruses on the application layer. However, the weakness is that it has a great impact on the overall performance of the system. The proxy server has to be set for all applications generated by the client. It greatly increases the complexity of system management.

4) Monitoring:
Monitoring firewalls with the technology are beyond the original firewall definition. The monitoring firewall can actively and instantly monitor the data of each layer. Based on analyzing these data, the monitoring firewall can effectively judge the illegal intrusion. However, the implementation cost of monitoring firewall is high and not easy to manage.

Integration of security technologies includes two aspects, the fusion of different security products and the integration of the network equipment and security products. The fusion of different security products comes from the source of network attacks, and the fusion of security technologies is used to counter the attacks. In the construction of network security, the intrusion detection technology has to focus on monitoring, controlling and early warning whereas the firewalls could play the role to access control. Although the antivirus software and the security certification belong to different products, they could function independently. Beyond discussing which one performs better, it is more practical to find how to integrate and to make these products working together effectively. In the fusion of security technology, intrusion prevention focuses on the detection and the firewalls focus on the access control [14]. However, due to the limitation of hardware and technology, the performance of the system could decline rapidly. Hence, the integration has become a frontier topic in the information security. Similarly, the fusion of firewall and antivirus has been lead to the product for the information security. Therefore, the cost of security products could be higher if the technologies focusing on different security concern are effectively integrated. The tendency of this convergence not only focuses on the security technologies, but also on the systems and architectures. The products of network security are no longer just security devices, but also supported by a network device or platform.

The integration of security products and network equipment is the combination of security concerns including the routers, switches, terminals and other products with a management software. A comprehensive network security system is constructed with the network controller of terminal, firewall, intrusion detection, traffic analysis and monitoring, and content filtering. This network transformation connects the product to the overall security system. It means the integration of network technology involves the business and the applications. Security companies cooperate with Internet companies in a more complex model. Hence, the integration of security products and network equipment constitutes a security system which ultimately forms a secure network that the users could trust.

Network security is an important topic regarding how to solve the problems in the network with the technologies and tools to maintain a highly reliable network. A highly strict management is the major role towards the network security. It should be realized that any network security and data protection precautionary measures have certain limitation. There is no security system existed forever. People have to work together for the network security continuously. In the following, NSSA model is introduced in section 2. The NSSA estimation and prediction is described in section 3. Section 4 gives the illustration to the major technologies of NSSA. The application of NSSA used in Internet of Things is given in section 5. Conclusions and the suggestions for the future research are given in the final.

2 Network Security Situation Awareness Model

With the rapid development of computer network, existing the various attacks and the security incidents arising frequently, the resulting in the network security issue becomes attractive. In addition, cyberspace has gradually become a new battlefield for the security games among countries. Under this situation, the related network security issues become much more complicated. Moreover, the level of security technology is requested for a higher challenge. In the past years, the security protection is based on the firewall, the intrusion detection and the virus detection. This protection is hold after detecting the attack behavior. On the other hand, the warning alarm generated directly without the risk estimation, this is another pre-protection scheme. However, it results a high rate of mistakes and is obviously unable to meet the needs in deed. Based on the failure of traditional security defense system, situation awareness (SA) has been gradually applied in the field of network security. It could comprehensively perceive the threat situation of network security.

NSSA was inspired by the SA of aero-traffic control and applied the concept of SA to the field of network security [42]. Franke *et al.* [13] argued that "SA is a state that can be achieved in the varying degrees." Based on this idea, it could be thought that the network situation awareness is a part of SA related to the environment in the networks. The other related researches have been carried out on the security situation in the network. The concept of NSSA has been studied gradually. However, a unified and comprehensive definition of NSSA has not yet been formed. Most of NSSA are the detailed explanations for Endsley's definition [10]. There were no specific explanations for the field of network security. One could identify the network attack behaviors from a large number of collected data. Then, he could integrate this information to conduct a real-time online estimation to monitor on network security to achieve overall control of the network and to reduce the risk of network security. It is the essential purpose of NSSA to construct a secure network.

Two fundamental models, the conceptual model of SA proposed by Endsley [10] and the functional model and data fusion model proposed by Bass [4], lead a foundation for the researches on NSSA. Upon this basis, a variety of different NSSA models are derived. Although these mod-

els have different names, their principles are similar. The SA process could be divided into four stages, the observation, the orientation, the decision-making, and the action. The model can be well adapted to the SA in the complex internets. Based on the fusion, Xiaowu Liu *et al.* [29] proposed a network security cognitive sensor control model with the characters of cross-layer architecture and cognitive loop. It could improve the interaction and cognitive ability among the different network layers. Besides, based on the analysis of model components and functions, the fusion algorithm is used to make accurate decisions on the heterogeneous multi-sensor security events. Combining with the reasoning of the relationship between the threat genes and the threat levels, a hierarchical quantification method including service layer, host layer and network layer is proposed. The advantage is that the method overcomes the insufficiency of dealing with the complex relationships among network components, and improves the ability to express network threats.

The framework of network SA research proposed by Zhenghu Gong *et al.* [16] is based on the Joint Directors of Laboratories (JDL) data fusion model. The model summarizes the content of network SA research. Comparing with the traditional model, this model highlights the nature of dynamic loop and continuous refinement with the importance on the feeding back. Combining the JDL data fusion model and Endsley's SA model, An *et al.* [1] extended and proposed a network SA model. The model consists of four layers, including the identification layer, the understanding layer, the prediction layer and the measure layer. Comparing with the three-layer structure in the traditional model, the more comprehensive model adds a measure layer which assists the decision makers to provide an optional valuation.

The NSSA system was proposed by Kokkonen [21]. It composed of the input connection layer, the information normalization layer, the data fusion layer and the visualization layer. The model emphasizes the role of visualization, including the human-computer interaction connection. With sharing the information between human and computer, it is more conducive to operate in the actual environment. Genghong Lu *et al.* [31] proposed an industrial control network SA model, and applied SA to the industrial control systems. From the bottom to the top layer, the model consists of three parts, the acquisition layer for the situation elements, the situational estimation layer and the subsequent SA process. It provides a new method for the security assurance in the industrial control systems.

Without the similarity of the above-mentioned hierarchical structure, the framework of network SA were proposed by Jajodia *et al.* [18] consists of a series of technologies and automation tools. It uses the automation tools to replace network analysis, and continuously understands the system by asking questions of the system, the security status and the impact and evolution of attacks. Most of the current models are based on the three-layer architecture. They are supplemented from the perspectives of dynamic looping, visualization, and automation. These models are enriched and refined according to the needs of different application scenarios. The above models could be the relatively classic SA models. In addition, scholars have proposed the different SA models for the different applications and scenarios. Azhagiri M.A. *et al.* [3] proposed a belief Markov multi-stage transferable model for the advanced persistent threats. The worth model approaches the accuracy in the NSSA. According to the above analysis, it is obvious that the most of the current NSSA models are based on the traditional three-layer model, supplemented with the different perspective improvements to meet the actual needs. With concerning the increasingly complex internet, it is necessary to innovatively improve the NSSA model to realize the intelligent perception.

3 Estimation and Prediction of Network Security Situation Awareness

According to the logical analysis framework of SA, this section explains the situation estimation and the situation prediction. The extraction of network security situation elements is the basis on the network security situation research. With the data preprocessing to delete the redundant data, to extract more important situational elements, and to standardize the original data, it provides a basis for the situational estimation and situational prediction [41]. Essentially, the acquisition of situational elements is used to classify the data in the network, and to determine whether each data is abnormal and to determine what kind of abnormality it belongs to. Network security situation estimation is mainly to obtain the various monitoring data in the network. It supports the network administrator to make decisions and to prepare for the protection according to the domain knowledge and the historical data with the network security feature. The comprehensive estimation for network security status is with the help of mathematical models. With integrating the macro network security situations, the network administrator could make the decisions and take protective measures in advance and provide the information for the next situation prediction [9]. The important role of network security situation estimation is to provide a strong support to implement for the security protection. Network security situation estimation is the important link to realize the NSSA system.

Security situation estimation is to focus the difficulty of NSSA and is lack of a systematic theory. The researches in the field of situational estimation are relatively scattered. Most of them have their own independent viewpoints. There is no unified method that could be used for the estimation. The methods and technologies to measure the quality of estimation are still relatively lack. It leads to the diverse estimation methods with the author-

itative consensus. Presently, there are many research results on the network security situation estimation. According to the theoretical and technical basis, it could include three categories, the estimation based on the mathematical models, the estimation based on the probability and knowledge reasoning, and the estimation based on the pattern classification. The mathematical model-based estimations mainly include the analytic hierarchy process (AHP), the set-pair analysis, and the distance deviation method. These estimations comprehensively consider the factors that affect the NSSA and, then, establish the corresponding relationship between the security index set and the security situation. The situation estimation is transformed into a multi-index comprehensive estimation or a multi-attribute collection problem [5]. There are an appropriate formula and a decision rule to gives the result. This type of method is the earliest and widely used estimation for NSSA. However, there are many estimations based on this method. The definition of the variables involves many subjective factors and is lack of the objective and unified standards.

The major proposed estimations are based on the probability and knowledge reasoning. These reasoning schemes include the fuzzy theory, Bayesian network, Markov theory, Dempster–Shafer theory, etc. These models are set up based on the expert knowledge and the experience database, and applying with the logical reasoning to make the security situation estimation. The main idea is to deal with the randomness of the network security events with the help of fuzzy theory and Dempster–Shafer theory. According to the expert knowledge and the databases of experience, these estimations use the logical reasoning to estimate the security situations. The idea is to deal the randomness of network security events with the help of fuzzy theory and Dempster–Shafer theory. Using these methods to build a model needs to acquire the prior knowledge first. In practical, the method for acquiring the knowledge is still relatively simple and mainly relying on the machine learning or the expert knowledge. However, the machine learning has the difficulties on the operation, and the expert knowledge relies on the accumulation of experience. Because of a large number of rules and knowledge needed and the much more complex reasoning process required, it is difficult to apply to a large-scale network for the estimation. The pattern classification based estimation includes the cluster analysis, the rough set, the grey analysis, the neural network and the support vector machine, etc. Based on the training model and the pattern classification, it estimates network security situation. This method with a good learning ability could establish a relatively mathematical model correctly. However, the amount of calculation is much large In practical applications, such as a long modeling time needed in the rough set and the neural network and the large number of incomprehensible features. It cannot be well applied in the real time network environment.

According to the above analyses, each method has its advantages with the applicable scenarios. On the other hand, there exists the difficulties for each estimation. Most scholars combine the multiple algorithms to achieve the optimization. Dong *et al.* [44] proposed a quantitative estimation based on Back Propagation (BP) neural network combined with Cuckoo searching. It Introduces the momentum factor and self-adjusting learning rate to optimize the algorithm. Also, it makes the convergence speed and the estimation accuracy improved. F. Li *et al.* [25] proposed an algorithm which combining Simulated Annealing algorithm and Baum Welch algorithm has a better performance than that of Markov model. It could be seen that the combination of multiple algorithms will be the tendency for the research in the future.

Network security situation prediction refers to the forecast of network security situation change trend in the network in the future based on the current network security situation estimation and the existing historical data. According to the recent researches, this paper sorts out the current popular situation forecasting, mainly including the neural network, the support vector machine and the time series forecasting, etc. [11]. Zhuo Ying *et al.* proposed a generalized regression neural network model and gave the network design principles of the model for network situation prediction [49]. They verified the accuracy and time efficiency of the model. Yicun Wang [38] proposed a prediction based on the fuzzy Markov to predict the threaten value in the network security. Wang Xiao *et al.* [37] proposed Markov time-varying model suitable for the real-time risk probability prediction. Liu Jie *et al.* [28] proposed a nonlinear network traffic prediction based on BP neural network, and Fan Julun *et al.* [12] proposed a network security prediction based on the radial basis function (RBF). Meng Jin *et al.* [33] used the hybrid hierarchical genetic algorithm (HHGA) to train the RBF neural network to improve the accuracy of prediction. Wei Bin *et al.* [39] studied the network security defense model based on the ensemble learning, and proposed a prediction with a better accuracy and a generalization ability with employing the ensemble learning algorithms. Kang HW *et al.* [20] predicted the urban security based on the two-column convolutional neural network with the context and target information.

Currently, the neural network is the most commonly used method for the network situation prediction. It has the characteristics of self-learning, self-adjustment and nonlinear processing. It is very suitable for the nonlinear complex systems. Also, it could approach the good results in the network security situation prediction. Commonly, the applied neural networks include BP neural network, RBF neural network, the feedback neural network and so on. However, it takes a long training time in using the neural network. Also, it has the problems of overfitting or undertraining, and the difficulties of providing the credible explanations. To improve these problems, it is often necessary to combine other algorithms for the optimization. Using the multiple groups of chaotic particles to optimize the key parameters of the gray neural network

would make the accuracy improvement.

Support Vector Machine (SVM) is a pattern recognition based on the statistical learning theory. The principle of SVM is to map the input space vector to a high-dimensional space through a nonlinear mapping. With performing the linear regression on the space, the nonlinear regression problem is transformed to that likes to solve a linear regression problem in the high-dimensional space. However, in terms of the parameter selection, it is necessary to combine other algorithms to optimize the parameters. With the analysis of support vector machine and the improved particle swarm optimization (PSO) algorithm, the accuracy of prediction for the network situation could be significantly improved [8].

The time series prediction is used to reveal the regulation of the situation changing with the historical data. With this regulation, the situation predictions are made for the future. The advantage is that it is simple, intuitive, convenient for the practical application. However, the disadvantage is that the order of appropriate model and the optimal model parameter estimation are required for the higher accuracy requirement. Also, the modeling process is complicated [47].

Deep learning is an effective training method for the neural networks. It is in the rapid development currently. The major models include Restricted Boltzmann Machine (RBM), Autoencoder (AE) Convolutional Neural Network (CNN), Deep Stacking Network (DSN), Loop Application Network, Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM). The feature extraction could be effectively obtained from a large amount of complex data by employing these learning models. It could solve the problem of the feature extraction and make the prediction for the network security situation. Currently, the problems needed to be solved in applying the deep learning methods to the network security include the performance of the algorithm, the interpretability, the traceability, the self-adjustment, the self-learning, the false alarm and the imbalanced datasets etc.

4 Major Technologies of Network Security Situation Awareness

The major technologies of NSSA include the SA methods based on the hierarchical analysis, the machine learning, the immune system and the game theory. Analytic Hierarchy Process (AHP) is a multi-objective and multi-criteria decision analysis method that combines the quantitative and qualitative analysis. The SA based on the hierarchical analysis handles the more complex process in three layers, i.e. service layer, host layer, and system layer. It Simplifies each layer, from the bottom to the up, and estimates the overall security situation of the system by calculating the local impact of the underlying security elements.

It is worth to refer Chen Xiuzhen *et al.* [7], the quantitative evaluation model of the hierarchical network system security threat situation. The model has four layers, the network system, the host, the service and the attack/vulnerability from the top to the bottom. The model is based on the massive alarm information of the intrusion-detection system (IDS) and the network performance indicators, and is the integration of the importance of services, the host and the organizational structure of the network system. However, there are some deficiencies in the model. The only source of safety information in this estimation is IDS alarm information. In practical, the actual network system deployment, such as the firewalls, the system logs, etc. are the indispensable security factors. If the information could not be included in the calculation, the advantages of network security situation evaluation technology in the comprehensive security information and the overall network security situation will be lost. Hence, some improved models are proposed. Lai Jibao [24] proposed an improved AHP that is constructed a multi-level and a multi-quantitative evaluation model of the network security situation. In this mode, the actual network system is decomposed into three layers, the network layer, the host layer, and the attack /defense layer from the top to the bottom, according to the scale and the hierarchical relationship. Using the multi-source security information provided by IDS, Firewall, VDS, etc. as the original data, it makes the more comprehensive information source. Zhang Yongchuan [46] proposed a three-layer model including the thematic layer, the element layer and the overall layer. Each layer evaluates the network security situation from the different perspectives with the different nuances. It constructs a multi-layer situation evaluation framework. With the proposed hierarchical model, the information sources could be more comprehensive and reliable to improve the authenticity and feasibility of SA in the real applications, obviously.

The characteristics of the security situation value is uncertain and nonlinear. Machine learning, having the good self-adjustment, the self-organization and the infinite approximation ability, performs excellently in describing the nonlinear complex systems. Currently, there are a lot of attentions on using machine learning for the situation evaluation and the situation prediction. Especially, Support Vector Machine (SVM) and Radial Basis Function (RBF) are used for the examples. There are three methods needed to discuss in the neural network and the wavelet neural network. Since SVM has a good generalization ability and is not easy overfitting, many researchers use SVM and Support Vector Regression (SVR) methods for the situation estimate and prediction. Although SVM has the advantages of fast convergence and strong anti-overfitting ability compared with the neural network does, using SVM alone as a prediction model comes with the problems of blind parameter selection in the training process. For the accuracy improvement of the prediction, Chen *et al.* [6] introduced the idea of regression prediction and proposed Random-Forest (RF)-SVM algorithm. The algorithm could predict the potential attacks to the data stream in the future network based on

the previous attack data. It not only improves the prediction accuracy, but also reduces the error.

RBF neural network has the advantages of strong function approximation, fast learning and self-adjustment ability. It could be used to describe the nonlinear and complex systems. Hence, it is suitable for the network security situation prediction. For example, the combination of RBF neural network and the time series prediction could well realize the prediction of network security situation. However, In practical applications, the neural network will have a character of slow convergence. It leads the difficulties to design the network layer and it is easy to fall into the problem of local optimization. Therefore, many researchers combine other schemes to optimize the parameters and the structure of RBF neural network. Jiang Yang *et al.* [19] proposed an improved particle swarm optimization (PSO) algorithm. Li Xixi [26] combined fuzzy c-means (FCM) algorithm and hybrid hierarchical genetic algorithm to Improve the learning process in the traditional restricted receptive field RRF deconvolution network.

For the same learning process, Wavelet Neural Network (WNN) has the characters of simpler structure, faster convergence, stronger learning ability and higher accuracy. While WNN approaches the global optimal, the local optimal could also be maintained. Because of these remarkable advantages, researchers begin to apply WNN to the field of SA. Also, they used the genetic algorithm, the population optimization algorithm, etc. to optimize WNN. Cong *et al.* [17] proposed the NSSA based on the optimized dynamic WNN. This scheme could combine the heterogeneous security data and the dynamic perception of the evolution tendency of the threats. It has the ability of self-regulation and control not only to achieve the goal of SA but also to provide a new method for the network monitoring and management.

Immunity is the ability of an organism to resist the infection. The problem of computer security systems is similar to the difficulty encountered in the biological immune systems. Because the biological immune system has the advantages of feature extraction, distributed detection, self-tolerance, self-adjustment and robustness, etc., and the ability of pattern recognition, learning and, memory, etc., it is suitable to applied to the SA research. Sun *et al.* [35] proposed a technique based on antibody concentration to describe the principles and framework of SA. They established a mathematical model with the lymphocyte life process. It makes the system could learn about the attack and location, severity, and the most severe areas of the intrusion are. Because of the disadvantages of poor scalability and the limited coverage of artificial immune systems, Qiao *et al.* [36] proposed the concept of collaborative artificial immune system and the related SA model. In the model, the memory detectors in the different computers can share the different points to improve the coverage and scalability of the artificial immune system. The technical combination of the artificial immunity and the cloud model monitors the network attacks using the intrusion detection based on the danger theory timely [45]. The network security situation could be evaluated with the changing of antibody concentrations. Besides, the situation could be predicted with employing the cloud model-based time series forecasting mechanisms.

Most of the traditional SA methods only focus on the attacking or defending. They usually ignore the situation of interdependent strategies on the both sides. In practical, in the actual attack and defense process, it is necessary to fully consider the possible opponent's strategies and formulate their own counter measurements. As long as there are offensive and defensive sides in the secure network, the game between these two will always be existed. Without considering the confrontation between the offensive and defensive sides in the secure network, it often leads to inaccurate results and greatly reduces the ability of SA model to describe the actual situation. Game theory is the study of mathematical theories and methods of struggle or competition. The concepts and characteristics of the two sides in the game are similar to those in the network security with the offender and defender. Researchers have been exploring the applicability of game theory to address the security issues in the internet. They applied the idea of game theory to NSSA. This issue has gradually become a popular research topic. With the game analysis in the behavior of threats among the administrators and the ordinary users, they established a three-party Markov game model to optimize and analyze the relevant algorithms and to make a real-time evaluation. Zhou *et al.* [48] proposed a NSSA method based on the preventing of threat propagation and set up the game model among the attackers, the defenders, and man-in-the-middle. For the real-time analysis, the system administrators could strengthen the most vulnerable nodes. For some existing evaluations without the comprehensive analyses of the threat, the protection, and the environmental information, the evaluation might lead an incorrect result. Weng Fangyu [40] proposed a network security situation estimation based on the attack/defense random game model. With establishing a threat propagation network and the stochastic game model, the proposed method processes the threats and defense to solve the difficulties Nash equilibrium mixed with the strategies. Based on the results of Nash equilibrium, the actions of the offensive and defensive sides are judged and the network security situation is quantitatively analyzed.

Visualization technology could be realized as the search of graphic information by expressing a large amount of abstract data with a visual form. It improves the efficiency of information processing in the visualization system. Kotenko *et al.* [23] proposed a visual analysis technique to display the security metrics that evaluates the overall network security situation and the efficiency of the protection mechanisms. The visualization technology of NSSA also has certain applications in the military for the real-time risk tracking. With utilizing a variety of data sources and methods, the visualized results are presented in the various types of graphs. The effect is more com-

prehensive and intuitive. Also, it helps the readers have a broader understanding of the current situation.

Dempster–Shafer theory by Dempste and Shafer is an important method for the data fusion and situation evaluation [27]. This method not only overcomes the inadequacy of describing uncertainty with probability but also has a flexible and changeable form. Dempster–Shafer theory could be combined with other methods such as the fuzzy logic, the neural network, and the expert system to further improve the accuracy of reasoning. For example, the traditional Dempster–Shafer theory could be improved by using the adaptive multi-swarm competition particle swarm optimization (AMCPSO) algorithm.

Liu Yuling *et al.* [30] proposed a network security situation prediction based on the spatial dimension analysis. It not only predicts the security situation element set in the future period but also analyzes the relationship within the security situation element set. Also, it predicts the impact on the network security situation. Liu's method considers the situation in the spatial dimension. On the other hand, Xi's scheme [43] deliberates on the network attribution. It could provide a more comprehensive analysis and the results are more accurate with combining the above two techniques. Technology, such as cloud computing and big data, provides new methods and ideas for SA. Saurez *et al.* [34] proposed Foglets which is a programming infrastructure for the geo-distributed computational continuum represented by the fog nodes and the cloud. It solves the problems existing in the application of the geographically distributed SA. The obtained information could be accomplished with the iterative analysis.

5 Application of Network Security Situation Awareness in IoT

The concept of SA from the human factors research in the aerospace flight originally. Thereafter, it has been widely used in the military, the air traffic supervision, the medical and other emergency dispatch fields. The NSSA is an important branch of SA. It could provide the operators with a comprehensive and reasonable decision support if the powerful global network monitoring and awareness capabilities are applied. With focusing on the application of SA in the field of network security, the following discusses from the perspectives of industrial control network and internet of things, respectively.

1) The integration of industrial control systems with internet has led to many threats and attacks. Also, it has a serious impact on the national security, the economic development and the social stability, etc. Hence, SA could be used for the effective monitoring and the overall control in the industrial system to ensure the safe operation. Energy Information Administration has built an automatic network intrusion and policy management system to achieve a real-time SA of the important infrastructure in the industrial systems in the USA. Lu *et al.* [32] proposed a NSSA of the industrial systems under the attacks based on the particle filtering and the voting mechanism to identify the malicious nodes It uses a security SA to provide an accurate situation evaluation for the system. Energy internet is a shared network that combines the energy technology and internet technology to achieve a high degree of integration of power, information, and business flows. Due to the important strategic position of power system, there are many studies devoted to the system. For the items and the measurement of the distributed energy security knowledge (DEnSeK), this method gives a detailed explanation. It could be seen that the application of NSSA in the smart energy internet will be an important development in the future.

2) Internet of things (IoT) has brought the third development of the global industry with the information technology. However, the security problems of IoT are becoming much more prominent. Kolbe *et al.* [22] proposed a context-based situation theory that can promote the framework of IoT SA with the knowledge base and the reasoning ability. Besides, IoT and wearable devices also offer new ideas for the healthcare. Anzanpour *et al.* [2] deployed a wireless local area network and used the wearable sensors to collect the patients' vital information. Based on the collected data, with leveraging data confidence evaluation, the knowledge base of patient health status, and the automatic reasoning to the health status, the patient's health status could be monitored effectively with carrying out overall SA. Vehicle-to-everything (V2X) is the combination of IoT and the self-organizing network of vehicles. Golestan [15] proposed the attention assisted framework to achieve SA of V2X. The framework uses a low-level data fusion and a high-level information fusion to realize the traffic entities, the situation, the impact evaluation and the decision-making, to achieve a safer and more complete V2X system.

Regarding the future development, NSSA should not be limited to the traditional applications. Adopting a variety of technologies and combining with the hierarchical detection, the potential security threats could be detected from the discrete and isolated data. Hence, with the security situation, the connection between the quick accurate complete and effective communication and the decision makers is still a very challenging problem.

6 Conclusion

This paper describes the major technologies of NSSA are given in the details from the aspects of the hierarchical analysis, the machine learning and the game theory, etc. The researches and the applications in the fields of NSSA are classified and summarized. The trend of development

and the application prospects of NSSA are expected and some problems that still need to be solved and may be faced in the future.

1) Functional module optimization:
According to the research on the data fusion, because of the wider applications of SA, the resulting data becomes huge and complex. The issue of the efficient and accurate process with these massive heterogeneous data is important. According to the research on situation prediction, with improving the accuracy of prediction and the forecasting ability of the network, it prevents the problems before the problem is hold. According to the research on the visualization of SA, with displaying the real time network status, it could provide an effective help for the decision-making.

2) Human-computer interaction and automatic response:
Currently, the network and system are still inseparable from the human participation. To establish a good human-computer interaction mechanism, indicating that the system accepts experts' suggestions and makes the modifications and adjustments, is both a future development trend and an urgent problem needed to be solved. Although people play an important role in the system, the system should be an independent entity. With facing the intrusion and attack, the system can not only alarm but also take certain protective measures to realize the automatic response and to improve the intelligence of the system.

3) Counter SA:
Counter SA refers to the concept that the attack and disrupt to the weaknesses or the flaws of SA systems, or employing other techniques to sabotage and interfere with the different stages of SA directly. For example, for the methods of situational evaluation and the situational prediction using the neural networks, the attackers might interfere with the training process of neural networks by adding malicious data, thereby affecting the accuracy of situation evaluation and prediction. In the severe cases, the diametrically opposite results may be measured and predicted and the system allows the attackers to take this advantage.

In general, the researches on SA is still at the initial stage. Especially, in the growing-up computer networks, there are still many treats and attacks needed to be improved. With the continuous improvement in the related technologies and researches, SA will definitely get greater development and give full play to its own advantages and characteristics to provide a strong guarantee to bring benefits to network security.

Acknowledgments

This work was partially supported by the Ministry of Science and Technology, Taiwan (ROC), under contract no.: MOST 107-2221-E-845 -002 -MY3 and MOST 110-2221-E-845 -002.

References

[1] J. An, X. Li, C. You, L. Zhang, "The research of cyber situation awareness model," in *Proceedings of International Conference on Intelligent and Interactive Systems and Applications*, pp. 232-238, 2016.

[2] A. Anzanpour, I. Azimi, M. Gotzinger, A. M. Rahmani, N. TaheriNejad, P. Liljeberg, A. Jantsch, N. Dutt, "Self-awareness in remote health monitoring systems using wearable electronics," in *Proceedings of IEEE Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2017.

[3] M. Azhagiri, A. Rajesh, S. Karthik, "A multi-perspective and multi-level analysis framework in network security situational awareness," *International Journal of Computer Networks and Communications Security*, vol. 5, no. 4, pp. 71, 2017.

[4] T. Bass, D. J. Gruber, "A glimpse into the future of ID," *The Magazine of USENIX & SAGE*, vol. 24, no. 3, pp. 40-49, 1999.

[5] P. Cao, E. Badger, Z. Kalbarczyk, R. Iyer, A. Slagell, "Preemptive intrusion detection: Theoretical framework and real-world measurements," in *Proceedings of the 2015 Symposium and Bootcamp on the Science of Security*, pp. 1-12, 2015.

[6] G. Chen, Y. Zhao, "RF-SVM based awareness algorithm in intelligent network security situation awareness system," in *Proceedings of 3rd Workshop on Advanced Research and Technology in Industry (WARTIA'17)*, 2017.

[7] X. Chen, Q. Zheng, X. Guan, *et al.*, "Quantitative assessment method of hierarchical network security threat situation," *Journal of Software*, vol. 17, no. 4, pp. 885-897, 2006.

[8] G. Dong, W. Li, S. Wang, X. Zhang, J. Lu, X. Li, "The assessment method of network security situation based on improved BP neural network," in *Proceedings of International Conference on Computer Engineering and Networks*, pp. 67-76, 2018.

[9] S. Dowling, S. Michael, M. Hugh, "Using analysis of temporal variances within a honeypot dataset to better predict attack type probability," in *Proceedings of 2017 12th International Conference for Internet Technology and Secured Transactions*, 2017.

[10] M. R. Endsley, S. J. Selcon, T. D. Hardiman, D. G. Croft, "A comparative analysis of SAGAT and SART for evaluations of situation awareness," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 42. no. 1. Pp. 82-86, 1998.

[11] C. Fachkha, E. Bou-Harb, A. Boukhtouta, S. Dinh, F. Iqbal, M. Debbabi, "Investigating the dark cyberspace: Profiling, threat-based analysis and correlation," in *Proceedings of 2012 7th IEEE International Conference on Risks and Security of Internet and Systems (CRiSIS)*, pp. 1-8, 2012.

[12] J. Fan, P. Wu, "Network security situation prediction method based on RBF neural network," *Journal of Xi'an University of Posts and Telecommunications*, vol. 22, no. 2, pp. 7-11, 2017.

[13] U. Franke, B. Joel, "Cyber situational awareness–a systematic review of the literature," *Computers & security*, vol. 46, pp. 18-31, 2014.

[14] Y. Gao, D. Li, Z. Cheng, "Research on situational awareness model of UAV distributed swarm," *Journal of Electronics and Information*, vol. 26, no. 2, pp. 301-325, 2000.

[15] K. Golestan, *Information Fusion Methodology for Enhancing Situation Awareness in Connected Cars Environment*, Ph.D. dissertation, University of Waterloo, 2016.

[16] Z. Gong, Y. Zhuo, "Research on Network Situational Awareness," *Journal of Software*, vol. 12, no. 7, pp. 1605-1619, 2010.

[17] C. Huang, C. Wang, "Network security situation awareness based on the optimized dynamic wavelet neural network," *International Journal of Network Security*, vol. 20, no.3, pp. 593-600, 2018.

[18] S. Jajodia, A. Massimiliano, "An integrated framework for cyber situation awareness," in *Theory and Models for Cyber Situation Awareness*, Springer, Cham, pp. 29-46, 2017.

[19] M. Jiang, Y. P. Luo, and S. Y. Yang, "Particle swarm optimization-stochastic trajectory analysis and parameter selection," in *Swarm Intelligence, Focus on Ant and Particle Swarm Optimization*, pp. 179-198, 2007.

[20] H. W. Kang, H. B. Kang, "Urban safety prediction using context and object information via double-column convolutional neural network," in *Proceedings of 2016 13th IEEE Conference on Computer and Robot Vision (CRV)*, 2016.

[21] T. Kokkonen, "Architecture for the cyber security situational awareness system," in *Internet of things, smart spaces, and next generation networks and systems*, Springer, Cham, pp. 294-302, 2016.

[22] N. Kolbe, A. Zaslavsky, S. Kubler, J. Robert, Y.L. Traon, "Enriching a situation awareness framework for IoT with knowledge base and reasoning components," in *International and Interdisciplinary Conference on Modeling and Using Context*, 2017.

[23] I. Kotenko, N. Evgenia, "Visualization of security metrics for cyber situation awareness," in *Proceedings of 2014 IEEE Ninth International Conference on Availability, Reliability and Security*, 2014.

[24] J. Lai, *Research on Several Key Technologies of Network Security Situational Awareness Based on Heterogeneous Sensors*, Ph.D. dissertation, Harbin Engineering University, 2009.

[25] F. Li, Q. Li, Z. Jiang, "Improved method of situation assessment method based on hidden Markov model," *Journal of Computer Applications*, vol. 37, no. 5, pp. 1331, 2017.

[26] X. Li, *Research on Network Security Situation Evaluation Based on Particle Swarm Neural Network*, Ph.D. dissertation, Hebei Normal University, 2018.

[27] L. Liang, S. Sun, M. Li, X. Li, "Data fusion technique for bridge safety assessment," *Journal of Testing and Evaluation*, vol. 47, no. 3, pp. 2080-2100, 2018.

[28] J. Liu, Y. Huang, "Nonlinear network traffic prediction based on BP neural network," *Computer Application*, vol.27, no. 7, pp. 1770-1772, 2007.

[29] X. W. Liu, H. Q. Wang, H. W. Lü, J. G. Yu, S. W. Zhang, "Fusion-based cognitive awareness-control model for network security situation," *Journal of Software*, vol. 27, no.8, pp. 2099-2114, 2016.

[30] Y. Liu, D. Feng, Y. Lian, "Network security situation prediction method based on spatiotemporal dimension analysis," *Computer Research and Development*, col. 51, no. 8, pp.1681-1694, 2014.

[31] G. Lu, D. Feng, "Situational awareness modeling of industrial control network based on particle filter," *Journal of Automation*, vol. 44, no.8, pp. 1405-1412, 2018.

[32] G. Lu, D. Feng, "Network security situation awareness for industrial control system under integrity attacks," in *Proceedings of 2018 21st IEEE International Conference on Information Fusion (FUSION)*, 2018.

[33] J. Meng, C. Ma, J. He, "Network security situation prediction model based on HHGA-RBF neural network," *Computer Science*, vol. 38, no. 7, pp. 70-72, 2011.

[34] E. Saurez, K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, "Incremental deployment and migration of geo-distributed situation awareness applications in the fog," in *Proceedings of the 10th ACM International Conference on Distributed and Event-based Systems*, 2016.

[35] F. Sun, X. Feng, "Antibody concentration based method for network security situation awareness," in *Proceedings of 2009 3rd IEEE International Conference on Bioinformatics and Biomedical Engineering*, 2009.

[36] F. Sun, X. Feng, "A network security situation awareness model based on cooperative artificial immune system," in *Proceedings of 2011 IEEE International Conference on Computer Science and Service System (CSSS)*, 2011.

[37] X. Wang, Q. Li, Y. Qi, "A real-time analysis method of network security risk based on Markov model," *Computer Science*, vol. 43, no.2, pp. 338-341, 2016.

[38] Y. Wang, W. Li, Y. Liu, "A forecast method for network security situation based on fuzzy Markov chain," in *Advanced Technologies, Embedded and Multimedia for Human-centric Computing*, pp. 953-962, 2014.

[39] B. Wei, M. Zhang, "Research on network security defense model based on ensemble learning algorithm," *Journal of Armed Police Engineering University*, vol. 33, pp. 69, 2017.

[40] F. Weng, *Research and Design of Network Security Situation Assessment and Prediction Method Based on Stochastic Game Model*, Ph.D. dissertation, Beijing University of Posts and Telecommunications, 2018.

[41] J. Wu, L. Yin, Y. Guo, "Cyber attacks prediction model based on Bayesian network," in *Proceedings of 2012 IEEE 18th International Conference on Parallel and Distributed Systems*, 2012.

[42] W. Wu, C.Y. Yang, "An overview on network security situation awareness in internet," *International Journal of Network Security*, vol.24, no.3, pp. 450-456, 2022.

[43] R. Xi, X. Yun, Y. Zhang, Z. Hao, "An improved quantitative evaluation method of network security situation," *Chinese Journal of Computers*, vol. 38, no.4, pp. 749-758, 2015.

[44] H. Zhang, Q. Huang, F. Li, J. Zhu, "A network security situation prediction model based on wavelet neural network with optimized parameters," *Digital Communications and Networks*, vol. 2, no. 3, pp. 139-144, 2016.

[45] R. Zhang, L. Tao, X. Xin, Y. Shi, "A network security situation awareness model based on artificial immunity system and cloud model," in *Proceedings of International Conference on Information and Management Engineering*, 2011.

[46] Y. Zhang, *Research and System Implementation of Network Security Situational Awareness Model*, Ph.D. Dissertation, University of Science and Technology of China, 2010.

[47] Y. Zhang, D. Zhao, J. Liu, "The application of Baum-Welch algorithm in multistep attack," *The Scientific World Journal*, 2014.

[48] B. Zhou, L. F. Zhong, "Network security situation awareness based on intercepting the threat spread," in *Proceedings of 2013 3rd IEEE International Conference on Computer Science and Network Technology*, 2013.

[49] Y. Zhuo, Q. Zhang, Z. Gong, "Generalized regressive neural network model for network situation prediction," *Journal of PLA University of Science and Technology (Natural Science Edition)*, vol. 13, no.2, pp. 147-151, 2012.

Biography

Wen Xi studied at Fujian Normal University from 2007 to 2010, earned a master of Science degree. In September 2010, joined Fuzhou University of International Studies and Trade, Worked at the Big Data Academy, the title is Lecturer.

Wei Wu studied at Chung Buk National University from 2008 to 2011, majored in Political Diplomacy, and obtained a master's degree. In 2011, entre Wuhan Digital Engineering Institute. The title is an engineer.

Cheng-Ying Yang received an M.S. degree in Electronic Engineering from Monmouth University, New Jersey, in 1991, and a Ph.D. degree from the University of Toledo, Ohio, in 1999. He is a member of the IEEE Satellite & Space Communication Society. Currently, he is employed as a Professor at the Department of Computer Science, University of Taipei, Taiwan. His research interests are performance analysis of digital communication systems, error control coding, signal processing, and computer security.

Research on a Trustworthy Digital Learning Roll Call System

Anthony Y. H. Liao[1], Yu-Ying Hsieh[2], Cheng-Ying Yang[3], and Min-Shiang Hwang[1,4]
(Corresponding author: Min-Shiang Hwang)

Department of M-Commerce and Multimedia Applications, Asia University[1]
Department of Computer Science and Information Engineering, Asia University[2]
No. 500, Lioufeng Rd., Taichung 41354, Taiwan (R.O.C.)
Department of Computer Science, University of Taipei[3]
Taipei 10048, Taiwan (R.O.C.)
Department of Medical Research, China Medical University Hospital, China Medical University[4]
No. 91, Xueshi Rd., Taichung 40402, Taiwan (R.O.C.)
Email: mshwang@asia.edu.tw
(Received Sept. 22, 2020; Revised and Accepted May 28, 2022; First Online June 6, 2022)

Abstract

Since the New Coronavirus Pneumonia (COVID-19) outbreak in 2019, it has seriously threatened people's health, life, and lifestyle. As a result, universities and colleges worldwide have changed their teaching methods from physical to digital teaching. However, this digital teaching model has generated many problems that have never existed before. For example, how to effectively ensure student participation in learning and monitor student attendance. In particular, the reliability of the digital roll call results has been questioned by many. In order to understand the above problems and explore ways to improve, this paper organizes and analyzes a large number of literature and research materials and attempts to propose a specification of a trusted digital learning roll call system that meets expectations. We also propose a guideline for developing and implementing the trusted digital teaching system.

Keywords: Digital Roll Call System; Digital Teaching; Learning Management Systems (LMS)

1 Introduction

The digital roll call system refers to the ability to provide students with online sign-in, automatic, or manual roll call during teaching activities. This system can help teachers grasp students' attendance and absence in real-time. Teachers can use various roll call methods on the digital learning platform. In addition to the methods mentioned above, teachers can also use an APP with a roll call function to assist in obtaining students' attendance. The primary purpose of implementing roll call is to enable teachers to grasp students' attendance and understand their learning status instantly. In addition to being used as a reference for teaching design, it can also promote students' attendance and participation in learning and can be used as a reference for learning guidance.

From the observation of literature and practice, it is generally believed that the roll call implementation will help students attend and improve their learning effectiveness. Therefore, the roll call mechanism is regarded as one of the critical tasks of instructional design. Furthermore, scholars Zhu *et al.* [42], after comparing physical and online courses, believe that student attendance is positively related to improving students' performance. Their analysis of the literature found that many studies have shown that attendance and participation are the main factors affecting student learning outcomes [12, 17, 20].

Higher test scores and better test scores are often associated with higher or lower attendance levels [9]. Nieuwoudt's research pointed out that the most significant impact on learning achievement is the time students learn in digital learning systems [30]. Therefore, most teachers attach great importance to the attendance of students. Especially in digital teaching, in order to prevent students from being absent or lazy, thus reducing their motivation for learning and making their learning performance poor, roll call is listed as one of the teaching design items [1].

The roll call method has also developed from the traditional pen and paper login to the browser-based digital roll call. Digital teaching is being implemented in large numbers. How to give full play to the motivation of guiding students to participate actively and learn independently to improve the reliability of digital roll call is a research topic that needs to be discussed in depth at this stage.

2 Literature Review

2.1 The Relationship between Student Attendance and Learning Outcomes

From past literature, Rogers [32] and Golding [19] used observational methods to investigate the correlation between student attendance and performance, and their findings were positive. Other experimental studies have pointed out that a clear attendance policy can improve student attendance and achievement [40]. Westerman *et al.* argue that attendance represents a particularly effective measure of behavior and an important factor affecting performance. After studying the relationship between attendance and performance in business management higher education classrooms, they found that attendance was positively correlated with test scores; the negative impact of absenteeism was more pronounced for lower-performing students than for higher-performing students; absence was associated with The cumulative average grades of students showed a negative correlation [38].

Most other studies on the relationship between attendance and academic performance have shown a strong relationship between student attendance and test scores. Scholars such as Brokowski & Dempsey [4], Chan *et al.* [6], Cohn & Johnson [10], and other scholars all believe that a mandatory attendance policy significantly reduces the absentee rate and improves students' test scores. Cohn & Johnson [10] compiled five-year student absence data in a higher education institution and found that the relationship between student attendance and learning performance was consistent throughout the five years. The above literature shows a significant relationship between students' learning effectiveness and attendance rate; the higher the attendance rate, the better the academic performance—conversely, the lower the attendance rate, the worse the learning effect. The research on the relationship between attendance rate and learning effectiveness is summarized in Table 1.

In order to understand whether there are differences in the attendance rate and test scores of students in different courses, Fadelelmoula [15] analyzed the impact of students' class attendance on their final exam scores in four courses in the second semester of the 2016-2017 academic year at Almaarefa College. Attendance in these courses is mandatory, and students must be above 75% attendance to sit for final exams. The study results found that attendance rates for all courses were positively correlated with final exam scores. In addition, Corbin *et al.* [11] found that students who regularly participated in class achieved higher scores in exams. Finally, Thomas & Higbee argue that although the strength of the relationship between attendance and test scores is debated, observations have shown a positive correlation between classroom attendance and student performance across multiple disciplines, including science, mathematics, physics, psychology, and information technology [36].

2.2 Reasons for Student Absenteeism Behavior

The above literature showed that "students' attendance rate in class is positively correlated with test scores". Student absenteeism has come under the spotlight since Covid-19 sparked a flood of online classes. Many scholars try to find out the reasons to find a solution to the problem. Huimin *et al.* [23] reviewed the literature and stated that autonomous motivation, controlled motivation, self-efficacy, and teaching quality are critical factors for college students to participate in courses. When any of the four factors is weak, students' motivation to participate is also low. Bulach [5] and Ghosh *et al.* [18] pointed out that students' trust in digital learning courses is an essential factor affecting their participation in courses. Trust in digital learning is defined as "the degree to which students are willing to rely on digital learning systems and how teachers or educational institutions take adequate measures to help students increase their confidence in using digital learning systems [5, 18]. Students' confidence in the digital learning management system, including the trust of teachers and schools, can influence attendance. In addition, students' trust in a teacher determines how students are willing to accept the teacher's teaching [39]. Therefore, trust is also a necessary condition for achieving good learning outcomes.

In addition to the viewpoints in the above literature, there are many reasons for college students to miss classes. For example, course type, learning motivation, number of students, influence by peers, class time, teacher factors, etc. [16]. Devadoss & Foltz [13] believe that students miss classes for various reasons. Absenteeism is a significant problem for many higher education institutions and a primary concern for educators. The study points out that two factors contribute to absenteeism: Background factors, such as learning patterns, background, employment, and the practicality of course content. The other is behavioral factors, including attitudes to participating in learning and personal characteristics [33]. Blerkom pointed out that the most common reasons cited by students for absenteeism were boredom, illness, clashing with other classes, or social activities [2]. The results of Chenneville and Jordan [7] showed that the reason for absenteeism is that many college students do not fully understand the impact of absenteeism on their grades. Bond [3] pointed out that students' participation in the classroom is affected by factors such as teachers, curriculum, technology, family, peers, and individuals. It can be seen from the above literature that scholars have different views on the reasons for college students' absenteeism. These articles are summarized in Table 2 according to the literature.

2.3 Related Research on Digital Roll Call Systems

In recent years, due to the impact of Covid-19, with the change from face-to-face to online teaching, the re-

Table 1: Research on the relationship between attendance rate and learning effectiveness

Research Scholars	Research Results
Zhu *et al.* [42]	The degree of rigor with which teachers set attendance and achievement standards is positively associated with improved student attendance and performance.
Gump [20], Dalelio [12], Gbadamosi [17]	Attendance and classroom participation are significant factors that affect student learning outcomes.
Clump *et al.* [9]	Higher test scores are often associated with higher or lower attendance rates.
Nieuwoudt*et al.* [30]	The most significant impact on learning achievement is the time students spend on the digital learning system.
Rogers [32], Golding [19]	The correlation between student attendance and academic performance is positive.
Westerman*et al.* [38]	Attendance rates are positively correlated with test scores. The negative impact of absenteeism was more pronounced for lower-performing students than for higher-performing students. Absence was negatively correlated with the student's cumulative performance average.
Brokowski & Dempsey [4], Chan *et al.* [6], Cohn & Johnson [10]	Mandatory attendance policies have significantly reduced absenteeism and improved student test scores.
Cohn & Johnson [10]	The relationship between student attendance and academic performance remained consistent over the five years.

Table 2: Reasons for college students' absenteeism

Research Scholars	Reasons for Students' Absenteeism
Huimin *et al.* [23]	Autonomous motivation, controlled motivation, self-efficacy, and teaching quality are critical factors for college students to participate in courses.
Bulach [5] and Ghosh, *et al.* [18]	Students' willingness to use digital learning management system factors, including teachers' and schools' trust, will influence attendance.
Wooten & McCroskey [39]	The student's trust in the teacher determines the degree to which the student is willing to accept the teacher's teaching.
Friedman *et al.* [16]	There are many reasons for college students to miss classes. For example, course type, motivation to learn, number of students, exposure to peers, class time, and teacher characteristics.
Sawon *et al.* [33]	Two factors cause absenteeism: First, background factors—for example, study mode, the background of origin, current employment, and practicality of course content. The second is behavioral factors, including attitudes toward participating in learning and personal characteristics.
Blerkom.[25]	The most common reasons students cited for absenteeism were boredom, illness, running with other classes, or attending social events.
Chenneville & Jordan. [7]	Reasons for truancy: Many college students do not fully understand the impact of truancy on their grades.
Bond. [3]	Student engagement in the classroom is influenced by teachers, curriculum, technology, family, peers, and individuals.

liability of digital roll call has become an issue that schools and teachers want to break through. Many studies have shown that in implementing digital teaching, although teachers use a variety of roll-call methods, each roll-call method has advantages and disadvantages [8, 14, 21, 22, 24–29, 31, 34, 35, 37, 41]. For example, scholars Tigang & Xiaodan [35] surveyed the current stage of university education. They concluded that teachers commonly use the following types of roll call methods: verbal roll call, sign-in or login, fixed seat mapping, online homework, photographing, fingerprint recognition, face recognition, blueprint Bud scanning, radio frequency identification (RFID), application software (APP), and QR code, and other methods.

Othman *et al.* [31] pointed out that the traditional approach is time-consuming, error-prone, and risks losing records. Thanks to the development and advancement of various networks, recording and reporting student attendance can already be fully automated. For example, the Interactive Student Attendance Management System (ISAMS) records student attendance through barcode scanning. The interaction of students and lecturers or the transfer of files can also be processed instantaneously [26], thus simplifying the process of attendance [22]. Long and Hao [27] proposed using visual programming (VB) to develop a roll call function software that randomly selects a student for roll call. And the roll call results will be automatically saved to the Excel table. It can make the final statistical work more accessible and more interesting, but it can also significantly reduce the burden on teachers. Islam *et al.* [24] designed an android application to collect student attendance and store it in a database. This attendance record is then emailed to students and parents. However, teachers still have to manually mark students' attendance records and store their roll call results in the database, consuming time and effort [29].

Mittal *et al.* [28] believe that the traditional online roll call may be replaced or fraudulent, whether it is student login, manual check-in, or scanning ID. Jayant and Borra [25] proposed the concept of a cloud intelligent roll call system. Feature detection and face recognition using the Viola-Jones object detection framework. But according to the actual measurement, the accuracy is only about 40% [28]. Chintalapati & Raghunadh [8] and Dongliang [14] proposed to combine face detection and face recognition to achieve more accurate roll call accuracy.

Many other scholars [37, 41] proposed the concept of combining Radio Frequency Identification (RFID) and the Internet of Things (Internet of Things, IoT) for online roll call. If RFID and IOT are combined and cloud storage is used, the read data can perform better [34]. For example, Guohui & Ruisheng [21] proposed an IoT service architecture consisting of five units, including users, RFID tags (Tag) attached to places and objects, RFID reading of mobile devices, internet, and back-end system. These units constitute an accurate digital roll call system.

2.4 Comprehensive Summary and Analysis

Based on the viewpoints of the above literature, the digital roll call system generally used by teachers of various universities at present can achieve the purpose of roll call. Still, it has different advantages and disadvantages regarding correctness and efficiency (as shown in Table 3). The selection of the appropriate roll-call method must be based on the organization's environmental planning and facilities and equipment as the basis for selecting the roll call method.

We can find from Table 3 that although the traditional oral roll call method is time-consuming and troublesome, it is the most accurate and reliable and can achieve the purpose and function of roll call. The roll call method of barcode scanning can simplify the roll call process and process the interaction between teachers and students or file transfer in real-time. The implementation process is simple and convenient, but individual scanning and manual inspection are required, and it is difficult to prevent students from cheating or replacing. Using visual programming (VB) software for roll call has the advantage that the roll call results will be automatically stored in an Excel table and can also be automatically counted at the end of the period. Still, they can only be randomly selected and cannot be used for simultaneous roll calls by multiple people, which is the most significant disadvantage. The way to use the android application to collect student attendance, the attendance record will be emailed to students and parents for the record. But teachers still have to manually mark students' attendance records and store the results in the database, wasting time and effort. The advantages of face detection and face recognition roll call are convenient, fast, time-saving, and not accessible for students to impersonate or replace. The disadvantage is that the recognition effect is affected by resolution and face changes, resulting in recognition errors. There is also the issue of student privacy. The roll call system combining RFID and IoT has the advantages of convenience, speed, and time-saving, and students are not easy to impersonate or replace. However, it is susceptible to interference or contamination, resulting in reading failures.

3 Results and Discussion

3.1 Advantages and Disadvantages of Digital Roll Call System

Due to the impact of Covid-19, the type of teaching has changed from face-to-face to digital courses in a hurry. How to correctly, effectively, and quickly make remote roll calls has become a research topic everyone is concerned about. Many scholars have invested in research, hoping to find the best method. In this paper, through various literature discussions and research results, the existing system does not have a convenient and effective roll call system,

Table 3: Comparison of different digital roll call systems

Roll calling methods	Advantages	Disadvantages
Verbal roll call [31, 35]	Accurate, reliable, and students cannot impersonate or cheat.	Time-consuming, hassle, and delays in-class time.
Barcode Scanning [22, 26]	Simplify the roll call process and handle real-time teacher-student interaction or file transfer.	Individual scans and manual inspections are required, making it challenging to prevent student fraud or impersonation.
Using Visual Programming (VB) software [27]	The roll call results will be automatically saved in an Excel spreadsheet. The student attendance rate can be automatically counted at the end of the semester.	The software can only randomly select one person but is not used for multiple people to take roll calls simultaneously.
Using an android app to collect student attendance [24]	Attendance records are then emailed to students and parents.	Teachers still have to manually mark students' attendance records and store the results in the database.
Feature detection and face recognition using the Viola-Jones object detection framework [25, 28]	Convenient, fast, time-saving, and not accessible for students to impersonate.	The recognition effect is affected by the resolution and face change, causing errors. There are also privacy concerns.
Combining face detection and face recognition [8, 14]	It can achieve the accuracy of roll call more accurately, conveniently, and quickly, save time, and it is not easy for students to impersonate.	The recognition effect is affected by the resolution and face changes caused by errors. There are also privacy concerns.
A roll call system combining RFID and IOT [21, 34, 37, 41]	Convenient, fast, time-saving, efficient, and not easy to an impostor.	Vulnerable to interference or smears, causing read failures.

is fast, and can prevent fraud. However, no matter which roll-call method is adopted, it has its shortcomings. Some roll call methods are fast and convenient but lack accuracy. Some roll call methods require additional software or hardware to perform. The common disadvantage of all roll call methods is that the roll call system cannot prevent fraud entirely.

3.2 Study Limitations

1) The literature sources collected in this article are mainly monographs, academic studies, journal papers, seminar papers, and Internet articles. However, since the practice has not been implemented, there are inevitably some omissions in the integrity of the data. Thus the research results cannot be used as comprehensive inferences.

2) From the literature, it is found that some views and designs are the author's conception, lack verification by empirical research, and it is not easy to judge their practicality.

3) This article only discusses the methods and effects of digital roll call. In addition, these studies only focus on the results of establishing a digital roll call mechanism and have not discussed more important administrative support, such as roll call policy, roll call system, software and hardware equipment, and information technology training for teachers.

3.3 Suggestions for Future Research

In response to the problems described in Subsection 3.2, it is suggested that we can add digital roll call methods such as practical observation and in-depth interviews to the research on related topics to understand the situation of implementing digital roll calls at the teaching site. At the same time, the roll call policy, roll call system, software and hardware equipment, and information technology training for teachers are included in the research scope to make the research results more complete.

The following are evaluation criteria for good and trusted digital learning roll call system:

1) Correctness: Accurately roll call classroom students to avoid impostors.

2) Privacy: Only students and lecturers know about the roll call behavior. Students or others not in the class cannot infer whether other students are present in the course.

3) Efficiency: The shorter the roll call time, the better it cannot affect class time.

4) Equipment: No additional hardware or software equipment is required. Many students own and use

mobile phones, so it is acceptable to use mobile phones or tablet computers as a trusted digital learning roll call system user device. Developing and using APP software is necessary, but students install the APP only once, not every class.

5) Verifiability: Attendance information must be carefully preserved. If future students doubt that the information is correct, there needs to be a mechanism to verify it.

6) Statistics and notification: Statistics students' attendance rate and notify students of attendance information.

How to develop a trusted digital learning roll call system that meets the above evaluation criteria is a future essential research topic.

4 Conclusion

The digital roll call aims to make up for the sense of alienation between teachers and students because they are located in different places. And promote students' active participation and active learning in order to achieve teaching goals. Therefore, digital roll call is essential when implementing digital course activities. This paper aims to explore the reliability of digital roll call and find a correct and effective roll-call method through the sorting and analysis of various literature.

The study results found that no matter how digital roll call was performed, satisfactory results could not be achieved. Accordingly, this paper proposes the viewpoint of "integrating roll call into teaching content". That is to say, teachers incorporate roll call into teaching activities in teaching design and use lively and interesting real-time question and answer, asking students to share exciting life stories, problem-solving experience, views on cases, etc., to replace the rigid and severe roll call process. This technique of making the name invisible can make the course more vivid and achieve the purpose of the roll call. It is a method of roll call that teachers and students are willing to accept.

Finally, we propose some evaluation criteria for designing an excellent and trustworthy digital learning roll call system. Future research must develop a trusted digital learning roll call system that meets these evaluation criteria.

Acknowledgments

The Ministry of Science and Technology partially supported this research, Taiwan (ROC), under contract no.: MOST 109-2221-E-468-011-MY3, MOST 108-2410-H-468-023, and MOST 108-2622-8-468-001-TM1.

References

[1] A. Agarwal, S. Sharma , V. Kumar, and M. Kaur, "Effect of e-learning on public health and environment during covid-19 lockdown," *Big Data Mining and Analytics*, vol. 4, no. 2, pp. 104–115, 2021.

[2] V. Blerkom, "Class attendance in undergraduate courses," *The Journal of Psychology*, vol. 126, 487-494, 1992.

[3] M. Bond, "Facilitating student engagement through the flipped learning approach in K-12: A systematic review," *Computers & Education*, vol. 151, 2020.

[4] W. Brokowski, R. Dempsey, "Attendance policies and student performance," *Clearing House*, vol. 53, no. 3, pp. 129–130, 1979.

[5] C. Bulach, "A measure of openness and trust," *People and Education*, vol. 1, pp. 382-392, 1993.

[6] K. Chan, C. Shum, D. Wright, "Class attendance and student performance in principles of finance," *Financial Practice and Education*, vol. 7, no. 2, pp. 58–65, 1997.

[7] T. Chenneville, C. Jordan, "Impact of attendance policies on course attendance among college students," *Journal of Scholarship of Teaching and Learning*, vol. 8, pp. 29-35, 2008.

[8] S. Chintalapati, M. V. Raghunadh, "Automated attendance management system based on face recognition algorithms," in *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC'13)*, 2013.

[9] M. A. Clump, H. Bauer, A. Whiteleather, "To attend or not to attend: Is that a good question?," *Journal of Instructional Psychology*, vol. 30, pp. 220-224, 2003.

[10] E. Cohn, E. Johnson, "Class attendance and performance in principles of economics," *Education Economics*, vol. 14, no. 2, pp. 211–233, 2006.

[11] Corbin, K. Burns, A. Chrzanowski, "If you teach it, will they come? Lay students' classattendance and student engagement," *Legal Education Review*, vol. 20, no. 1 & 2, pp. 13-44, 2010.

[12] C. Dalelio, "Student participation in online discussion boards in a higher education setting," *International Journal on E-Learning*, vol. 12, pp. 249-271, 2013.

[13] S. Devadoss, S. Foltz, "Evaluation of factors influencing student class attendance and performance," *American Journal of Agricultural Economics*, vol. 78, pp. 499-507, 1996.

[14] H. Dongliang, S. Linxiang, G. Wenjun, "A study of roll call system for teachers based on RFID," in *IEEE 4th International Conference on Computer Science and Information Technology (ICCSIT'11)*, vol. 2, 2011.

[15] T. Fadelelmoula, "The impact of class attendance on student performance," *International Research Journal of Medicine and Medical Sciences*, vol. 6, no. 2, pp. 47-49, 2018.

[16] P. Friedman, F. Rodriguez, J. McComb, "Why students do and do not attend classes: myths and realities," *College Teaching*, vol. 49, no.4, pp. 124-133, 2014.

[17] G. Gbadamosi, "Should we bother improving students' attendance at seminars?," *Innovations in Education and Teaching International*, vol. 52, no. 2, pp. 196-206, 2015.

[18] A. K. Ghosh, T. W. Whipple, G. A. Bryan, "Student trust and its antecedents in higher education," *Journal of Higher Education*, vol. 72, no. 3, pp. 322-340, 2001.

[19] J. M. Golding, "The role of attendance in lecture classes: you can lead a horse to water," *Teaching of Psychology*, vol. 38, no. 1, pp. 40-42, 2011.

[20] S. E. Gump, "The cost of cutting class: Attendance as a predictor of success," *College Teaching*, vol. 53, no. 1, pp. 21-26, 2011.

[21] J. Guohui, W. Ruisheng, "Research on RFID based IoT-Take the management of cultural and creative goods in N Museum as an example," in *Proceedings of Taiwan Internet Symposium*, pp.1-6, 2013.

[22] O. Harumi, M. Mito, M. Masami, "Development of the online attendance management system," *Memoirs of Nishinippon Institute of Technology*, vol. 36, pp. 111-114, 2006.

[23] L. Huimin, H. Pijung , L. Uden, Y. Changho, "A multilevel investigation of factors influencing university students behavioral engagement in flipped classrooms," *Computers & Education*, vol. 175, 2021.

[24] M. M. Islam, M. K. Hasan, M. M. Billah, M. M. Uddin, "Development of smartphone-based student attendance system," in *HumanitarianT echnology Conference (R10-HTC'17)*, IEEE, pp. 230–233, 2017.

[25] N. K. Jayant, S. Borra, "Attendance management system using hybrid face recognition techniques," in *Conference on Advances in Signal Processing (CASP'16)*, pp.412-417.2016.

[26] S. Jonathan, S. N. Junaini, S. L. Lau, "ISAMS: Tracking student attendance using interactive student attendance management system," in *Proceeding of 3rd Malaysian Software Engineering Conference*, pp. 218-223, Malaysia: Selangor, 2007.

[27] Z. Long, W. Hao, "A design of a random roll call system based on VB," in *IEEE Second International Conference on Information and Computing Science*, vol. 1, pp. 316–318, 2009.

[28] A. Mittal, F. S. Khan, P. Kumar, T. Choudhury, "Cloud based intelligent attendance system through video streaming," in *IEEE International Conference On Smart Technology for Smart Nation*, pp. 1352-1357, 2017.

[29] L. Mothwa, J. R. Tapamo, T. Mapayi, "Conceptual model of the smart attendance monitoring system using computer vision," in *IEEE 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS'18)*, PP. 229-234, 2018.

[30] J .E. Nieuwoudt , "Investigating synchronous and asynchronous class attendance as predictors of academic success in online education," *Australian Journal of Educational Technology*, vol. 36, no. 3, pp. 15-25, 2020.

[31] M. Othman, S. N. Ismail, M. I. Md. Raus, "The development of the web-based Attendance Register System (ARS) for higher academic institution: From feasibility study to the design phase," *International Journal of Computer Science and Network Security*, vol. 9, no. 10, pp. 203-207, 2009.

[32] J. R. Rogers, "A panel-data study of the effect of student attendance on university performance," *Australian Journal of Education*, vol. 45, no. 3, pp. 284-295, 2001.

[33] K. Sawon, M. Pembroke, P. Wille, "An analysis of student characteristics and behaviour in relation to absence from lectures," *Journal of Higher Education Policy and Management*, vol. 34, no. 6, pp. 575-586, 2012.

[34] T. Sharma, S. L. Aarthy, "An Automatic Attendance Monitoring System using RFID and IOT using Cloud," in *IEEE Online International Conference on Green Engineering and Technologies (IC-GET'16)*, 2016.

[35] J. Tigang, Z. Xiaodan, "A survey for information technology based roll call methods in university education," in *IEEE 8th International Conference on Information Technology in Medicine and Education*, pp. 410-413, 2016.

[36] P. V. Thomas, J. L. Higbee, "The relationship between involvement and success in development algebra," *Journal of College Reading and Learning*, vol. 30, no. 2, pp. 222-232, 2000.

[37] M. A. Vouk, "Cloud Computing Issues, Research and Implementations," in *Journal of Computing and Information Technology*, vol. 16, no. 4, pp. 235-246, 2008.

[38] J. W. Westerman, L. A. Perez-Batres, B. S. Coffey, R. W. Pouder, "The relationship between undergraduate attendance and performance revisited: alignment of student and instructor goals. Decision Sciences," *Journal of Innovative Education*, vol. 9, no. 1, pp. 49-67, 2011.

[39] A. G. Wooten, J. C. McCroskey, "Student trust of teacher as a function of socio-communicative style of teacher and socio-communicative orientation of student," *Communication Research Reports*, vol. 13, no. 1, pp. 94-100, 1996.

[40] Z. Yizhong , Y. Fuhao, L. Jingren, X. Junyuan, "Online roll call and sign-in system," in *International Symposium on New Media Design, Communication and Technology Application*, 2008.

[41] S. Zhang, X. Chen, X. Huo, "Cloud computing research and development trend," in *Proceedings of the Second International Conference on Future Networks*, pp. 93-97, 2010.

[42] L. Zhu, E. Huang, J. Defazio, S. A. Hook, "Impact of the stringency of attendance policies on class attendance/participation and course Grades," *Journal of the Scholarship of Teaching and Learning*, vol. 19, no. 2, pp. 130-140, 2019.

Biography

Anthony Y. H. Liao received his M.S. degree in computer science and the Ph.D. degree in computer science and engineering both from the University of Louisville, Louisville, Kentucky, USA. He is currently an associate professor and the Chairman of the Department of M-Commerce and Multimedia Applications, Asia University, Taiwan. Dr. Liao is a senior member of IEEE, and a member of ACM. His research interests include artificial intelligence, image processing, pattern recognition, data mining, e-learning, management information systems, enterprise resource planning, e-commerce, smart manufacturing, and software engineering.

Yu-Ying Hsieh received the Bachelor of Education in National Taiwan Normal University in 1983 and received an M.S. of Social Welfare in Providence University, Taiwan, in 2003. She served as a kindergarten teacher and principal from 1982 to 2005. She has repeatedly won the Kindergarten Model Award and Social Welfare Service Award issued by government agencies from 1996 to 2003. She was currently studying for a Ph.D. in the Department of Computer Science and Information Engineering of Asia University. Her research interests include applying information technology in-game and event design, etc.

Cheng-Ying Yang received the M.S. degree in Electronic Engineering from Monmouth University, New Jersey, in 1991, and Ph.D. degree from the University of Toledo, Ohio, in 1999. He is a member of IEEE Satellite & Space Communication Society. Currently, he is employed as a Professor at Department of Computer Science, University of Taipei, Taiwan. His research interests are performance analysis of digital communication systems, error control coding, signal processing and computer security.

Min-Shiang Hwang received M.S. in industrial engineering from National Tsing Hua University, Taiwan in 1988, and a Ph.D. degree in computer and information science from National Chiao Tung University, Taiwan, in 1995. He was a distinguished professor and Chairman of the Department of Management Information Systems, NCHU, during 2003-2011. He obtained 1997, 1998, 1999, 2000, and 2001 Excellent Research Awards from the National Science Council (Taiwan). Dr. Hwang was a dean of the College of Computer Science, Asia University (AU), Taichung, Taiwan. He is currently a chair professor with the Department of Computer Science and Information Engineering, AU. His current research interests include information security, electronic commerce, database, data security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 300+ articles on the above research fields in international journals.

Research on Network Traffic Data Anomaly Identification and Detection Based on an Intrusion Detection Algorithm

Hui Zhang

(Corresponding author: Hui Zhang)

Yancheng Teachers University

No. 50, Kaifang Avenue, Yancheng, Jiangsu 224002, China

Email: psy7hz@yeah.net

(Received Feb. 23, 2020; Revised and Accepted May 28, 2022; First Online June 23, 2022)

Abstract

Network traffic data anomalies can pose a significant challenge to the network's security, and accurate identification and detection of these anomalies are required. In this paper, based on the intrusion algorithm, an improved Synthetic Minority Oversampling Technique (SMOTE) method was used to balance the data volume. Then the traffic features were selected by random forest (RF). The improved gray wolf optimization-back propagation neural network (IGWO-BPNN) algorithm was obtained by optimizing the parameters of BPNN with the improved GWO algorithm. Experiments were performed on the UNSW-NB15 data set. It was found that the algorithm had the highest accuracy when the top 13 features in RF ranking were used as input; the IGWO-BPNN had the best performance compared with other intrusion detection algorithms, with an accuracy (ACC) of 97.83%, a false negative rate (FPR) of 1.28%, and a false positive rate (FNR) of 3.07%. Furthermore, it had balanced accuracy in identifying different types of data. The experimental results verify the reliability of the proposed method and provide a new method for the identification and detection of network traffic data anomalies.

Keywords: Anomaly; Identification And Detection; Intrusion Detection; Network Traffic Data; Neural Network

1 Introduction

With the rapid development of technology, the emergence of big data and artificial intelligence has made both individuals and governments store data on the network more and more. Along with the rapid growth of network traffic data, more and more serious threats have emerged in the field of network security. The development of technology has also led to a dramatic increase in the number of network attacks and more diverse and sophisticated attack methods [16], which, in serious cases, can bring huge losses to individuals and enterprises. In this case, it is particularly important to identify and detect anomalies in network traffic data, which is of great importance for the healthy operation of the network, and therefore, intrusion detection algorithms have been increasingly studied [2].

Maleh *et al.* [7] proposed a support vector machine (SVM)-based approach for wireless sensor networks (WSNs), combined with a set of signature rules, and the simulation results showed that the method was able to perform better detection of anomalies in the network. Rastegari *et al.* [12] designed a genetic algorithm-based method, conducted experiments on data sets such as NSL-KDD, and found that the method provided a simple and readable set of rules and had a good detection performance. Aljawarneh *et al.* [1] used a genetic algorithm based on the voting algorithm with information gain to filter the data, combined classifiers such as J48, Meta Pagging, and RandomTree to form a hybrid algorithm for detection, and found that the algorithm showed higher accuracy. Hosseini *et al.* [5] designed a method combining multi-criteria linear programming and the particle swarm algorithm and found from the results on KDD CUP 99 that the method had advantages in terms of detection rate and running time. Based on the UNSW-NB15 data set, this paper designed an improved neural network method and also studied the data volume balance, feature selection, etc. The experiments found that the designed method had a good performance for the identification and detection of network traffic data anomalies. The method can be further promoted and applied in practice and makes some contributions to the development of network security.

2 Network Traffic Data Anomalies and Data Processing

Any damage, modification or access without authorization can be considered an intrusion into the network. Network anomalies can be characterized by traffic characteristics. If the traffic significantly deviates from the normal value, it can be considered a network traffic data anomaly. Several common types of network traffic data anomalies are as follows.

1) Worm virus [8]: it is a virus that spreads through emails, vulnerabilities, etc. Its possible hazards include consuming network bandwidth, causing network paralysis, stealing important information, damaging infrastructure, etc.

2) Scan attack: it scans the host to find network vulnerabilities and generates a large number of packets in a short period of time, causing a blockage in the network.

3) Distributed denial of service attack (DDoS) [4]: the attacker takes advantage of a flaw in the network and sends a large number of messages to the target through multiple hosts or servers, causing the target host to run out of resources and unable to provide normal requests.

4) Remote communication: the attacker uses a specific server prepared in advance to connect with the attacked network to do monitoring, data transmission and other actions.

Intrusion detection can detect unauthorized, illegal behaviors [10], which can be divided into two types, misuse intrusion detection, and anomaly intrusion detection. Misuse intrusion detection detects based on pattern matching. It detects the system and determines whether the system is intruded through the establishment of a pattern library of abnormal behaviors, but this method can only detect intrusions that already exist in the pattern library and has a poor detection capability for unknown attacks. Anomaly intrusion detection determines the intrusion or not by learning the normal behavior pattern of the network. If the current behavior has a large deviation from the normal behavior in the pattern library, it is regarded as an intrusion. The flow of anomaly intrusion detection is shown in Figure 1.

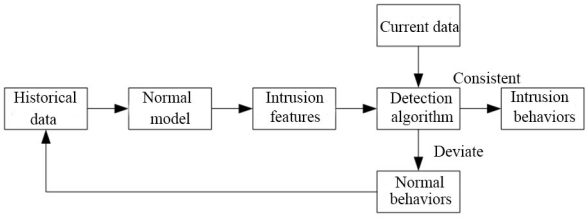


Figure 1: Anomaly intrusion detection algorithm

Before intrusion detection, two problems need to be dealt with first. One is that the normal traffic is significantly larger than the abnormal traffic in the actual collected data and there is an imbalance in the volume of abnormal data of different types. In order to improve the performance of the intrusion detection algorithm, it needs to be trained on a large amount of data. Therefore, the data needs balance. This paper uses the sampling-based method. The principle of the Synthetic Minority Oversampling Technique (SMOTE) [17] is to expand the samples of the minority class by synthesizing new samples through the calculation of the inter-sample distances. First, the Euclidean distance between every sample and other sample points in the minority class is calculated to obtain k nearest neighbor samples; then, the number of generated samples is determined according to the preset sampling ratio. For sample x in the minority class, assuming that the selected nearest neighbor is x', the new sample constructed can be written as:

$$x_{new} = x + rand\,(0,1) \times (x' - x)\,.$$

However, this approach may lead to data overlap. To address the above problem, SMOTE is improved. In the minority class, for every sample r_i $(i = 1, 2, \cdots)$, the closest m samples are found from the whole data set, and the samples of other classes are denoted by m'. If $m = m'$, i.e., the surrounding samples of r_i are all samples of other classes, then r_i is recorded as noise data, and this sample is not used in the generation; if $0 \le m' \le m$, i.e., not all the surrounding samples of r_i are samples of other classes, then r_i is supplemented according to SMOTE to reduce sample overlap.

The second problem is feature selection. Network traffic often has a large number of features. If all the features are used in the detection algorithm, it will greatly increase the computing time; therefore, in order to reduce the amount of computation, it is necessary to select suitable features. Random forest (RF) is used to select features in this paper.

RF has a good adaptability to unbalanced data sets [14]. The importance of features is determined by comparing their contributions to the three. It is assumed that there are m features in a data set, they can be divided into k types, the percentage of type k in m features is p_{mk}, the importance score of the i-th feature based on the GINI index is VIM_j^{GINI}, then the GINI index is: $G_m = 1 - \sum_{k=1}^{|k|} p_{mk}^2$; the importance score of feature x_i in m features is $VIM_{jm}^{GINI} = G_m - G_1 - G_r$, where G_1 and G_r are GINI indexes of new nodes after branching; the importance score of feature x_i on the j-th tree is: $VIM_{ji}^{GINI} = \sum_{m \in M} VIM_{im}^{GINI}$; the importance score of feature x_i in the whole RF is: $VIM_i^{GINI} = \sum_{j=1}^{n} VIM_{ji}^{GINI}$.

Finally, the degree of importance of every feature is obtained by normalization: $VIM_i = VIM_i / \sum_{j=1}^{m} VIM_j$. These features are ranked, and the top-ranked features are selected as the features for intrusion detection.

3 Intrusion Detection Algorithm Based on BPNN

A back-propagation neural network (BPNN) has strong adaptive and fault tolerance [18] and has good applications in data classification and prediction [11], which is the most used kind of neural network at present. However, BPNNs are influenced by the initial weights and thresholds, so an improved gray wolf optimization algorithm is used in this paper to optimize the initial values of BPNNs.

The gray wolf optimization (GWO) algorithm is a population intelligence algorithm [13] that mimics the hunting style of wolf packs. The population is divided into four levels, α, β, δ and ω, according to social relations, and α is the optimal solution. Let the distance between the gray wolf and the prey be D, the number of iterations be t, the prey position and the gray wolf position be X_p^t and X^t, then the encirclement process of the wolf pack can be written as: $D = |CX_p^t - X^t|$, and $X^{t+1} = X_p^t - AD$. C is the float factor, and A is the convergence factor: $C = 2r_1$, $A = 2ar_2 - a$, and $a = 2 - 2 \times \frac{t}{T}$, where r_1 and r_2 are the random numbers between 0 and 1, and a is the control parameter, which decreases linearly from 2 to 0.

In the GWO algorithm, parameter a is linearly decreasing. In order to further optimize the algorithm, this paper improves a with a sine function-based method. The calculation formula of the improved a is:

$$a = 2 \times \sin[\frac{\pi}{2} \times (\frac{t}{T} + 1)].$$

Moreover, the position update formula is also optimized through inertia weight φ, and the formulas after optimization are:

$$\begin{aligned} D_\alpha &= |C_1 X_\alpha - X^t|, \\ D_\beta &= |C_2 X_\beta - X^t|, \\ D_\delta &= |C_3 X_\delta - X^t|, \\ X_1^{t+1} &= \varphi X_\alpha^t - A_1 D_\alpha, \\ X_2^{t+1} &= \varphi X_\beta^t - A_2 D_\beta, \\ X_3^{t+1} &= \varphi X_\delta^t - A_3 D_\delta, \\ X^{t+1} &= \frac{X_1^{t+1} + X_2^{t+1} + X_3^{t+1}}{3}, \\ \varphi^{t+1} &= \varphi_{max} - \frac{t}{T}(\varphi_{max} - \varphi_{min}), \end{aligned}$$

where D_α, D_β and D_δ are distances between individuals α, β and δ and the current individual, X_α^t, X_β^t and X_δ^t are positions of individuals α, β and δ, φ_{max} is 0.9, and φ_{min} is 0.4.

The IGWO-BPNN algorithm is obtained by optimizing the parameters of BPNN with the improved GWO algorithm. The detailed steps are as follows.

1) The BPNN structure is initialized, and the number of nodes in every layer is determined.

2) The gray wolf pack is initialized.

3) The error of intrusion detection is regarded as the fitness function of the IGWO algorithm: $f = \sum_{i=1}^{N} |y_i - y_i'|$, where N is the number of nodes in the input layer, y_i is the actual output of the data set, and y' is the training output of the model.

4) The fitness is sorted. The position of the wolf pack and the parameters of the IGWO-BPNN algorithm are updated until it reaches the maximum number of iterations.

5) A model is built using the obtained weight and threshold, and experiments are conducted using the data in the test set.

4 Experimental Results

4.1 Experimental Setup

The operating system used for the experiment was Windows 10 (64 bit). The CPU was 2.80 GHz Intel Core i7. The memory was 16 GB. The NVIDIA GeForce MX 450 graphics card was used to speed up the operation, the programming language used was Python 3.6, and the model frameworks were TensorFlow and Keras. The data set used was UNSW-NB15 [9] and included 2,540,044 data, including 9 attack types. Every data had 49 features, as shown in Table 1.

The data sets used for the experiment are shown in Table 2.

It was seen from Table 2 that there was an imbalance in this data set. In the training set, there were many normal samples, and the number of worms samples was the least, 130. Therefore, the samples were expanded by the improved SMOTE method, and the expanded training set is shown in Table 3.

4.2 Evaluation Indicators

The performance of the algorithm was evaluated using the confusion matrix, as shown in Table 4.

The detailed indicators are as follows.

1) Accuracy: $ACC = \frac{TP+TN}{TP+TN+FP+FN}$.

2) False positive rate: $FPR = \frac{FP}{FP+TN}$.

3) False negative rate: $FNR = \frac{FN}{TP+FN}$.

4.3 Analysis of Results

First, the feature selection of the data set was analyzed. All the features of UNSW-NB15 were ranked in order of importance by RF, and the top 10-20 features were selected as the input of the IGWO-BPNN algorithm. The accuracy of the algorithm is shown in Figure 2.

It was seen from Figure 2 that the accuracy of the IGWO-BPNN algorithm was 96.79% when the top 10 features were used as input; the accuracy of the algorithm

Table 1: Characteristics of the UBSW-NB15 data set

Feature class	Feature description	Feature class	Feature description
Flow features	srcip	Time features	sjit
	sport		djit
	dstip		stime
	dsport		ltime
	proto		sintpkt
Base features	state		dintpkt
	dur		tcprtt
	sbytes		synack
	dbytes		ackdat
	sttl	General purpose features	is_sm_ips_ports
	dttl		ct_state_ttl
	sloss		ct_flw_http_mthd
	dloss		is_ftp_login
	service		ct_ftp_cmd
	sload	Connection features	ct_srv_src
	dload		ct_srv_dst
	spkts		ct_dst_ltm
	dpkts		ct_src_ ltm
Content features	swin		ct_src_dport_ltm
	dwin		ct_dst_sport_ltm
	stcpb		ct_dst_src_ltm
	dtcpb	Labelled features	attack_cat
	smeansz		Label
	dmeansz		
	trans_depth		
	res_bdy_len		

Table 2: Experimental data set

	Training set	Test set
Normal	56000	37000
Fuzzers	18184	6062
Analysis	2000	677
Backdoors	1746	583
DoS	12264	4089
Exploits	33393	11132
Generic	40000	18871
Reconnaissance	10491	3496
Shellcode	1133	378
Worms	130	44
Total	175341	82332

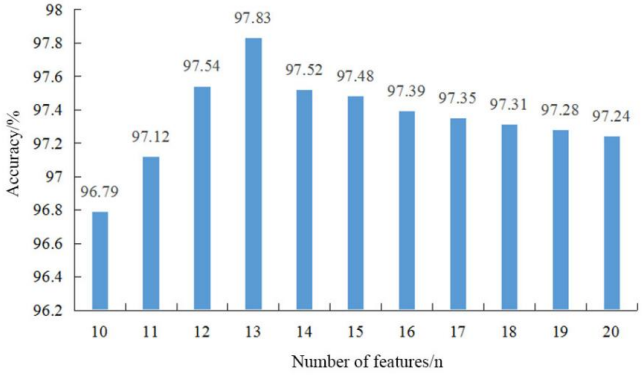


Figure 2: Accuracy of the algorithm under different numbers of features

gradually increased with the increase of the number of features; the accuracy of the algorithm was the highest (97.83%) when the number of features was 13; the accuracy of the algorithm decreased as the number of features continued to increase; the accuracy of the algorithm was only 97.24% when the top 20 features were used as input. Therefore, in feature selection, the top 13 features were used as the input of the algorithm to achieve a good accuracy rate. Then, to further demonstrate the performance of the designed algorithm, it was compared with decision tree (DT) [3], SVM [15], extreme learning machine (ELM) [6], and BPNN. The performance of different algorithms is shown in Table 5.

Table 3: Samples in the training set after expansion

	Training set after expansion by the improved SMOTE method
Normal	56000
Fuzzers	55325
Analysis	56214
Backdoors	55454
DoS	57245
Exploits	56325
Generic	55214
Reconnaissance	56256
Shellcode	55687
Worms	56528

Table 4: Confusion matrix

Actual class	Model output	
	Attack	Normal
Attack	TP	TN
Normal	FP	FN

It was seen from Table 5 that the accuracy of DT was 88.64%, the accuracy of BPNN was 93.54%, which was 4.9% higher than that of DT, and the accuracy of the IGWO-BPNN algorithm was 97.83%, which was 4.29% higher than that of the BPNN algorithm; the FPR of the IGWO-BPNN algorithm was 1.28%, and the FPR of the other algorithms was larger than 5%; the FNR of the IGWO-BPNN algorithm was 3.07%, and the FNR of the other algorithms was larger than 5%. It was concluded that the IGWO-BPNN algorithm performed best in identifying and detecting network traffic data anomalies. Finally, the accuracy of the IGWO-BPNN algorithm for different types of data was analyzed, and the results are shown in Figure 3.

It was seen from Figure 3 that after data balance, the amount of data of different types became less different, so the difference in accuracy was also less; the accuracy of

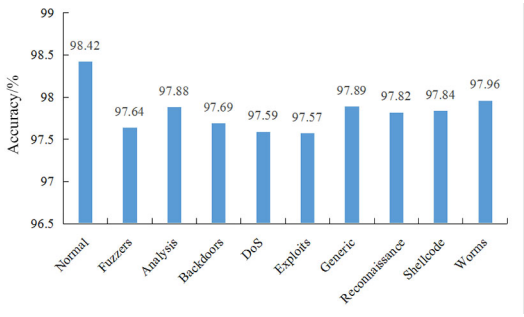


Figure 3: Accuracy of the IGWO-BPNN algorithm for different types of data

the IGWO-BPNN algorithm was highest for Normal data, reaching 98.42%, and lowest for Exploits data, reaching 97.57%; the accuracy of the algorithm for all types of data was above 97%, and the average accuracy was 97.83%. The above results verified that the IGWO-BPNN algorithm was reliable in detecting intrusions and could recognize and detect intrusion behaviors in the network effectively.

5 Conclusion

This paper studied the intrusion detection algorithm for the recognition and detection of network traffic data anomalies. The problem of unbalanced data volume was solved by using the improved SMOTE method. The features were selected by RF. The BPNN algorithm was improved and optimized to form an IGWO-BPNN algorithm. Experiments were conducted on the UNSW-NB15 data set. It was found that the accuracy of the IGWO-BPNN algorithm was the highest, 97.83%, when the number of features was 13; the IGWO-BPNN algorithm performed better in ACC, FPR, and FNR than the other intrusion detection algorithms and performed stably in detecting different types of attacks. The IGWO-BPNN algorithm can be further promoted and applied in practice.

References

[1] S. Aljawarneh, M. Aldwairi, M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *Journal of Computational Science*, vol. 25, no. MAR., pp. 152-160, 2017.

[2] A. Buczak, E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153-1176, 2017.

[3] D. R. Dadsena, "Rotational moment shape feature extraction and decision tree based discrimination of mild cognitive impairment conditions using MR image processing," *Biomedical Sciences Instrumentation*, vol. 57, no. 2, pp. 228-233, 2021.

Table 5: Performance comparison of different algorithms

	DT	SVM	ELM	BPNN	IGWO-BPNN
ACC	88.64%	90.12%	92.19%	93.54%	97.83%
FPR	17.83%	15.33%	10.97%	5.64%	1.28%
FNR	19.21%	14.38%	11.26%	8.33%	3.07%

[4] A. Ghaben, M. Anbar, I. H. Hasbullah, S. Karuppayah, "Mathematical approach as qualitative metrics of distributed denial of service attack detection mechanisms," *IEEE Access*, vol. 9, pp. 123012-123028, 2021.

[5] B. Hosseini, B. Amiri, M. Mirzabagheri, Y. Shi, "A new intrusion detection approach using PSO based multiple criteria linear programming," *Procedia Computer Science*, vol. 55, pp. 231-237, 2015.

[6] B. Li, H. Chen, T. Tan, "PV cell parameter extraction using data prediction–based meta-heuristic algorithm via extreme learning machine," *Frontiers in Energy Research*, vol. 9, pp. 693252., 2021.

[7] Y. Maleh, A. Ezzati, Y. Qasmaoui, M. Mbida, "A global hybrid intrusion detection system for wireless sensor networks," *Procedia Computer Science*, vol. 52, no. 1, pp. 1047-1052, 2015.

[8] A. Mondal, A. K. Das, S. Nath, R. T. Goswami, "Review study on different attack strategies of worm in a network," *Webology*, vol. 17, no. 2, pp. 363-375, 2020.

[9] N. Moustafa, J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Military Communications and Information Systems Conference (MilCIS'15)*, pp. 1-6, 2015.

[10] J. Peng, K. R. Choo, H. Ashman, "User profiling in intrusion detection: A review," *Journal of Network & Computer Applications*, vol. 72, pp. 14-27, 2016.

[11] G. Pi, L. Xu, Q. Ye, P. Hu, "Load prediction and control of capillary ceiling radiation cooling panel air conditioning system based on BP neural network," *IOP Conference Series: Earth and Environmental Science*, vol. 769, no. 4, pp. 042006 (7pp), 2021.

[12] S. Rastegari, P. Hingston, C. P. Lam, "Evolving statistical rulesets for network intrusion detection," *Applied Soft Computing*, vol. 33, pp. 348-359, 2015.

[13] M. R. Shakarami, I. F. Davoudkhani, "Wide-area power system stabilizer design based on Grey Wolf Optimization algorithm considering the time delay," *Electric Power Systems Research*, vol. 133, no. Apr., pp. 149-159, 2016.

[14] Q. Shu, T. Hu, S. Liu, "Random forest algorithm based on GAN for imbalanced data classification," *Journal of Physics Conference Series*, vol. 1544, pp. 012014, 2020.

[15] G. Sreedevi, B. Anuradha, "Feature extraction and classification of ECG signals with support vector machines and particle swarm optimisation," *International Journal of Biomedical Engineering and Technology*, vol. 35, no. 3, pp. 242, 2021.

[16] Y. Yu, J. Long, Z. Cai, "Network intrusion detection through stacking dilated convolutional autoencoders," *Security & Communication Networks*, vol. 2017, pp. 1-10, 2017.

[17] Y. D. Zhang, Y. Zhang, P. Phillips, Z. Dong, S. Wang, "Synthetic minority oversampling technique and fractal dimension for identifying multiple sclerosis," *Fractals*, vol. 25, no. 04, pp. 1740010, 2017.

[18] W. Zhu, H. Wang, X. Zhang, "Synergy evaluation model of container multimodal transport based on BP neural network," *Neural Computing and Applications*, vol. 33, no. 2, pp. 1-9, 2021.

Biography

Hui Zhang, born in 1977, has received the master's degree from Jiangsu University in June 2008. He is an associate professor and is working in Yancheng Teachers University. He is interested in software engineering theory and practice, network security monitoring.

A Pairing-free Identity-based Cryptosystem Using Elliptic Curve Cryptography

Poonsuk Ponpurmpoon and Pipat Hiranvanichakorn
(Corresponding author: Poonsuk Ponpurmpoon)

Graduate School of Applied Statistics, National Institute of Development Administration
148 Seri Thai Rd, Khlong Chan, Bang Kapi District, Bangkok 10240, Thailand
Email: poonsuk.ppp@hotmail.com, pipat@as.nida.ac.th
(Received Nov. 4, 2020; Revised and Accepted Jan. 16, 2022; First Online June 23, 2022)

Abstract

ID-based cryptosystems (IBCs) allow publicly identifiable information as public keys, which reduces the overhead of certificate management and eliminates the need for a certificate authority in the public-key infrastructure. Up to now, bilinear pairing technology is often used in the ID-based paradigm. However, it is expensive in computation time and is unsuitable for mobile network computing. Thus, interest in pairing-free ID-based algorithms is growing among researchers. We report a pairing-free IBC consisting of ID-based encryption, digital signatures, and key exchange schemes. All schemes use the same public and private key definitions, efficiently implementing IBC. Proof of the correctness and security analyses of the scheme are provided. Furthermore, comparative analysis with pairing-based and other pairing-free cryptosystems is discussed.

Keywords: Elliptic Curve Cryptography; Elliptic Curve Digital Signature Algorithm; Identity-Based Cryptosystems; Pairing-Based Algorithms; Pairing-Free Algorithms

1 Introduction

In 1984, Shamir [8] introduced the concept of the asymmetric-key ID-based cryptosystem (IBC) paradigm. In theory, the advantages of IBCs are that they allow the use of publicly identifiable information as public keys, which reduces the overhead cost of certificate management and gets rid of the need for a certificate authority (CA) in the public-key infrastructure. Public keys based on the user's identity are a meaningful and true reflection of the user's personality, while the user's public keys in non-IBCs are mathematically computed numbers.

Nowadays, the IBC concept is applied in many areas. For example, ID-based encryption (IBE) and ID-based signature (IBS) have been used for encrypting and signing messages between users in a hierarchical architecture for cloud computing (HACC) scenario [9]. The identity of the user is defined by nodes in the hierarchical structure from the user's node to the root node of the HACC. IBC has been utilized to generate the user's public key from his/her identity in an ID-based blind signature approach for E-voting [10]; IBC was utilized because it has the merit that the voter's public key is directly derived from his/her identity. IBC has been used to implement a dynamic authenticated group key agreement in [11]. In [12], anonymous ID-based broadcast encryption with asymmetric bilinear pairing was implemented.

In 2000, Joux [7] proposed the one-round three-party key agreement protocol that offers the potential of pairing with an ID-based paradigm. In 2001, Boneh and Franklin [3] introduced the first implementation of an IBE scheme by applying bilinear pairing on an elliptic curve for an IBC. Later, Smart [13] introduced the ID-based two-party key agreement protocol based on Weil pairing. However, as mentioned by Hu *et al.* [14], bilinear pairing is expensive in computation time and is considered to be unsuitable for mobile network computing. As Liu *et al.* [15] pointed out, bilinear groups with a large composite order do not provide substantial benefit to cryptographic schemes in practice. There is another important issue for an ID-based pairing algorithm. Since the public key from an IBC is derived from the user's identity, if the user's private key is compromised, it is not easy for the key generation center (KGC) to change the user's private and public key pairs. Thus, interest in pairing-free ID-based algorithms is growing, and several approaches to implement them have been proposed.

Many approaches using elliptic curve cryptography (ECC) have been suggested for ID-based pairing-free cryptosystems. In a digital signature scheme, Hu *et al.* [14] suggested a protocol based on the elliptic curve digital signature algorithm (ECDSA) [1]. They claimed that the computational time for the proposed protocol was lower than bilinear pairing ones. The disadvantage of this protocol is that the user's private key is not confidential because a part of the private key has to be sent out for use in the signature verification process. Nevertheless,

some pairing-free key agreement schemes have been proposed [4,16]. In the approach of Kumar and Tripathi [16], the key exchange parties must send out some parts of the private key to establish a shared key, and thus the private key is not confidential, and the scope of the research was also limited to parties that are members of the same KGC. However, in practice, shared keys established for users belonging to different KGCs are needed.

A pairing-free IBE algorithm has not yet been established, which may be due to the confidentiality problem in the use of the user's private key, as discussed above. Moreover, in previous ID-based pairing-free systems comprising digital signatures and key agreement, researchers have often defined the system parameters and private key extraction mechanism to fit their proposed protocols. Indeed, the use of the general system parameter and key definitions applied in the ID-based digital signature, encryption, and key exchange schemes in a cryptosystem to make its implementation simple has not previously been reported.

In this paper, a pairing-free IBC consisting of IBE, ID-based digital signature, and ID-based key exchange schemes is proposed. All of the schemes use the same public and private key definitions that have two advantages to address security concerns. First, the definitions can prevent other parties from discovering the KGC master key. Second, if the user's private key is compromised, the KGC can easily generate a new one. As for ID-based key exchange, the proposed system can cope with system parameters from different KGCs, which means that the users can be on different KGCs. This non-shared system parameter system is useful for mobile network computing in real scenarios.

We conducted a security analysis of the proposed system to address security concerns. The results show that it is durable to several types of attacks, such as man-in-the-middle, which can occur during the encryption-decryption and key exchange processes, and fake signatures, which can occur in the digital signature creation-verification process. Furthermore, we compared our proposed pairing-free scheme with some well-known pairing-free and pairing-based ones.

The remainder of this paper is organized as follows. In Section 2, we introduce the basic concepts of IBC, ECC, and bilinear pairing in cryptography. At the end of Section 2, some examples of pairing IBE algorithms, signature algorithms, and key exchange are offered. In Section 3, some examples of pairing-free algorithms and their security analyses are given. In Section 4, the proposed pairing-free algorithm and proof of its correctness are presented. The security and performance analysis of the proposed cryptosystem is discussed in Section 5. Finally, conclusions on the study are offered in Section 6.

2 Background for the Study

In this section, we briefly introduce four topics required as background for the study. First, the encryption and signature methods in IBCs are described followed by an overview of ECC. Next, we introduce the concept of bilinear pairing along with some of its properties. Finally, we review some pairing-based algorithms.

2.1 IBCs

IBCs first came to light in 1984 when Shamir [8] proposed a system that uses the user's identity information (e.g., email address, name, phone number, etc.) as a public key. This information is publicly known, so it does not require a CA to certify the key, which means that a special process to broadcast, store, and maintain the key is not required. In ID-based systems, a trusted authority such as a KGC generates the user's private key by using the user's public key and the KGC's private key. The KGC then sends the private key to the corresponding user via a secure channel.

2.1.1 IBE

In asymmetric-key cryptography, a message is encrypted by using the authenticated public key of the receiver, and its mathematical-related private key of the receiver is used to decrypt the message. In IBE, a message is encrypted by using the receiver's ID and the KGC public key. A detailed description of the IBE scheme can be found in [17]. A simple version of IBE and decryption is as follows.

Ciphertext C of message M is computed as

$$C = Encrypt(recipient_{pub}, KGC_{pub}, M),$$

where $recipient_{pub}$ is the public key of the recipient and KGC_{pub} is the KGC public key. The recipient uses his/her private key to decrypt the message. In general, the recovery process can be presented as

$$M = Decrypt(C, recipient_{pri}),$$

where $recipient_{pri}$ is the corresponding private key of the recipient.

2.1.2 IBS

A digital signature is a mathematical process to guarantee that the sender of the message cannot deny that he/she created the message, while the receiver cannot deny having received the message. It is created by applying the message to the signing algorithm. Details of IBS can be found in [5]. The signing and verification process can be presented as follows.

The signer uses his/her private key to sign the message. The user's signature is computed as

$$S_i = Sign(system\ parameters, signer_{pri}, M).$$

As for the verification process, the verifier uses the signer public key and KGC's public key to verify the signature. The logic for this is: If Verify(system parameters, KGC_{pub}, M, $signer_{pub}, S_i$), then accepts the signature.

2.2 ECC

ECC is very useful in several areas of mathematics. It is a new direction away from existing cryptosystems and plays an important role in ID-based cryptography in both pairing-enforced and pairing-free schemes [6]. In this subsection, we introduce some concepts of ECC that are of interest in our work. In ECC, we define the elliptic curve over finite field F_p, in the general form

$$E : y^2 = x^3 + ax + b,$$

where a and b are real numbers and $4a^3+27b^2 \neq 0$. Points on elliptic curve E and the point at infinity O, which is also an identity element in a finite Abelian group with the + operation, can be combined to produce a finite Abelian group with suitable properties. The following are some of the main properties of ECC.

2.2.1 Scalar Point Multiplication

$$kP = (P + P + P + \cdots + P)_{k\ times},$$

where k is a scalar and P is a point on elliptic curve E defined by the addition of the point k times. Some of the complex problems to do with ECC are as follows.

2.2.2 The Elliptic Curve Discrete Logarithm Problem (ECDLP)

Let E be an elliptic curve over finite field F_p and P be a point on elliptic curve E. For a given kP, find integer k. In ECDLP, kP is relatively easy to compute. However, computing k is intractable even when P and kP are known. In this problem, k is usually used as the user's private key, while kP is used as the corresponding public key.

2.2.3 The Elliptic Curve Diffie-Hellman Problem (ECDHP)

Let E be an elliptic curve over finite field F_p, P be a point on elliptic curve E, and a and b be integers. For a given $A = aP$, $B = bP$, find point $C = abP$.

In ECDHP, it is relatively easy to compute C when we know P, a, and B (or P, b, and A). However, exponential time is needed to compute abP even when P, A, and B are known. In this problem, abP is usually used as the shared key between users A and B.

2.3 Bilinear Pairing in Cryptography

In this subsection, we describe the basic theory of bilinear paring in cryptography, after which we address the properties of bilinear pairing. An overview of research into bilinear pairing can be found in [6].

Let G_1 be a cyclic additive group and G_2 be a cyclic multiplicative group. Both G_1 and G_2 are of prime order q. We define bilinear paring as mapping $\hat{e} : G_1 \times G_1 \rightarrow G_2$, which satisfies the following properties.

Computability. For all $U, V \in G_1$, there is an algorithm to efficiently compute $\hat{e}(U, V)$.

Non-degeneracy. There exists $P \in G_1$, such that $\hat{e}(P, P) \neq 1$.

Bilinearity. For all $U, V, W \in G_1$, $\hat{e}(U, V + W) = \hat{e}(U, V) \cdot \hat{e}(U, W)$ and $\hat{e}(U+W, V) = \hat{e}(U, V) \cdot \hat{e}(W, V)$. For $a, b \in Z_q$, $\hat{e}(aU, bV) = \hat{e}(U, bV)^a = \hat{e}(aU, V)^b = \hat{e}(U, V)^{ab} = \hat{e}(abU, V) = \hat{e}(U, abV)$.

2.4 Examples of Pairing-based IBCs

In this subsection, we provide some examples of pairing-based IBCs to compare the performance between pairing-based algorithms and our proposed pairing-free ones. The notation used in pairing-based schemes is defined in Table 1.

2.4.1 IBE Algorithms

The first practical IBE scheme on pairing was proposed by Boneh and Franklin [3]. Their encryption scheme applied pairing to compute the symmetric component between the sender and the recipient. In this algorithm, $sk_1 \in Z_q^*$ is the master key from the KGC, random number r $\in Z_q^*$, P is the group generator of G_1, G_1 is an additive group of points on E/F_p, G_2 is a multiplicative group of finite field $F_{p^2}^*$. $H_1 : (0,1)^* \rightarrow G_1^*$, $H_2 : G_2 \rightarrow (0,1)^n$. $Q_{ID} = H_1(ID)$. The user's private key $S_{ID} = sk_1 \cdot Q_{ID}$ and KGC's public key $P_{pub} = sk_1 \cdot P$, $g_{ID} = \hat{e}(P_{pub}, Q_{ID}) \in G_2^*$.

$$\begin{aligned} C &= (r \cdot P, M \oplus H_2(g_{ID}^r)). \\ M &= M \oplus H_2(g_{ID}^r) \oplus H_2(\hat{e}(r \cdot P, S_{ID})) \\ &= M \oplus H_2(g_{ID}^r) \oplus H_2(\hat{e}(r \cdot P, sk_1 \cdot Q_{ID})) \\ &= M \oplus H_2(g_{ID}^r) \oplus H_2(\hat{e}(P, Q_{ID})^{r \cdot sk_1}) \\ &= M \oplus H_2(g_{ID}^r) \oplus H_2(\hat{e}(sk_1 \cdot P, Q_{ID})^r) \\ &= M \oplus H_2(g_{ID}^r) \oplus H_2(\hat{e}(P_{pub}, Q_{ID})^r) \\ &= M \oplus H_2(g_{ID}^r) \oplus H_2(g_{ID}^r). \end{aligned}$$

2.4.2 IBS Algorithms

In 2002, Paterson [18] proposed an ID-based digital signature scheme based on bilinear pairing. The pairing is utilized to validate the genuineness of signature (R, Z) of message M. In this algorithm $k \in Z_q^*$ is a random integer and P is the group generator of G_1. $H_3 : (0,1)^* \rightarrow Z_q$, $H_4 : G_1 \rightarrow Z_q$, $Q_{ID} = H_1(ID)$, $S_{ID} = sk_1 \cdot Q_{ID}$, $sk_1 \in Z_q$ is the master key of KGC, and $P_{pub} = sk_1 \cdot P$ is the corresponding public key of KGC.

Signature (R, Z) is calculated as

$$\begin{aligned} R &= k \cdot P \\ Z &= k^{-1}(H_3(m) \cdot P + H_4(R) \cdot S_{ID}). \end{aligned}$$

Table 1: The notation used in pairing-based schemes

Notation	Type	Description
G_1	Finite field	An additive group of points on E/F_p
G_2	Finite field	A multiplicative group of finite fields $F_(p^2)^*$
sk_1	Integer	The KGC master key
P	A point on the elliptic curve	The group generator of G_1
Q_{ID}	A point on the elliptic curve	Th hash value of the user's ID
S_{ID}	A point on the elliptic curve	The user's private key
P_{pub}	A point on the elliptic curve	The KGC public key
(R, Z)	A pair of points on the elliptic curve	Signature (R, Z) of the user
K_A^B, K_B^A	$\in G_2$	The shared key between users A and B

The verifier can verify received signature (R,Z) by computing $\hat{e}(R,Z)$ and $\hat{e}(P,P)^{H_3(m)} \cdot \hat{e}(sk_1P, Q_{ID})^{H_4(R)}$. If $\hat{e}(R,Z) = \hat{e}(P,P)^{H_3(m)} \cdot \hat{e}(sk_1P, Q_{ID})^{H_4(R)}$, then the received signature (R, Z) is genuine. The following computation shows how we can use bilinearity to demonstrate the verification of received signature (R, Z).

$$\begin{aligned}
\hat{e}(R,Z) &= \hat{e}(k \cdot P, k^{-1}(H_3(m) \cdot P + H_4(R) \cdot S_{ID})) \\
&= \hat{e}(P, (H_3(m) \cdot P + H_4(R) \cdot S_{ID}))^{k \cdot k^{-1}} \\
&= \hat{e}(P, (H_3(m) \cdot P + H_4(R) \cdot S_{ID})) \\
&= \hat{e}(P, (H_3(m) \cdot P)) \cdot \hat{e}(P, H_4(R) \cdot S_{ID}) \\
&= \hat{e}(P, H_3(m) \cdot P) \cdot \hat{e}(P, \cdot S_{ID})^{H_4(R)} \\
&= \hat{e}(P, H_3(m) \cdot P) \cdot \hat{e}(P, sk_1 \cdot Q_{ID})^{H_4(R)} \\
&= \hat{e}(P,P)^{H_3(m)} \cdot \hat{e}(P, Q_{ID})^{H_4(R) \cdot sk_1} \\
&= \hat{e}(P,P)^{H_3(m)} \cdot \hat{e}(sk_1 \cdot P, Q_{ID})^{H_4(R)}.
\end{aligned}$$

2.4.3 ID-based Key Exchange

This is the last part of an IBC that uses a pairing technique. Smart [13] implemented the first ID-based key exchange protocol based on pairing. To obtain a shared key, $K_A^B = \hat{e}(aQ_B, P_{KGC}) \cdot \hat{e}(S_A, T_B)$ is computed for user A and $K_B^A = \hat{e}(S_B, T_A) \cdot \hat{e}(bQ_A, P_{KGC})$ is computed for user B.

In this algorithm, sk_1 is the secret key of the KGC; a and b are the session private keys of users A and B, respectively; $Q_{ID} = H_1(ID)$; the long-term private keys of the users are $S_A = sk_1Q_A$ and $S_B = sk_1Q_B$; and the session public keys of the users are $T_A = aP$ and $T_B = bP$.

The following proof shows that $K_A^B = K_B^A$.

$$\begin{aligned}
K_A^B &= \hat{e}(aQ_B, P_{KGC}) \cdot \hat{e}(S_A, T_B) \\
&= \hat{e}(aQ_B, sk_1 \cdot P_) \cdot \hat{e}(sk_1 \cdot Q_A, bP_) \\
&= \hat{e}(Q_B, P_)^{sk_1 \cdot a} \cdot \hat{e}(Q_A, P)^{sk_1 \cdot b} \\
&= \hat{e}(sk_1Q_B, aP) \cdot \hat{e}(bQ_A, sk_1 \cdot P) \\
&= \hat{e}(S_B, T_A) \cdot \hat{e}(bQ_A, P_{KGC}) \\
&= \hat{e}(bQ_A, P_{KGC}) \cdot \hat{e}(S_B, T_A) \\
&= K_B^A.
\end{aligned}$$

Even though several ID-based algorithms are based on bilinear pairing, the high computational complexity to compute bilinear pairings causes the computation time to be too long for actual implementation [2,14]. Another drawback of pairing-based ID-based schemes arises when the user's private key has been exposed. In ID-based schemes, the user's public key is the user's identity comprising his/her name, email address, phone number, etc. The user's identity does not often change, which means that his/her public key seldom does either. As the user's private key is generated by the corresponding public key, the private key is not easily renewed.

For the practical implementation of IBCs, a pairing-free basis has been considered to achieve better computation time. In general, pairing-free-based algorithms can be computed three times faster than pairing-based ones [2]. Hence, pairing-free schemes are more suitable for devices with limited power, such as mobile devices. In the next section, we introduce some pairing-free ID-based algorithms and an analysis of their security.

3 Security Analysis of the Current Pairing-free ID-based Schemes

Two pairing-free IBCs are discussed in this section. The first is the ID-based digital signature without pairing scheme proposed by Hu *et al.* [14]. The second one is the ID-based group key agreement without pairing scheme proposed by Kumar and Tripathi [16]. However, to the best of our knowledge, there are no pairing-free IBE algorithms. The notation used in the pairing-free schemes described in Sections 3 and 4 is defined in Table 2.

3.1 ID-based Digital Signature without Pairing [14]

In this algorithm, the KGC's private key is $sk_1 \in Z_n^*$, and $r_A \in Z_n^*$ is randomly produced to generate a private key for user A. The public key for user A is the user's ID and the corresponding private key is (D_A, s_A); D_A and s_A are respectively computed by the KGC as $D_A = r_A \cdot P$ and $s_A = (r_A + h \cdot sk_1) \bmod n$, where $h = HF_1(ID_A||D_A)$ is

Table 2: Notation for the pairing-free schemes

Notation	Type	Description
sk_i	Integer	The private key of KGC_i, for $i = 1, 2$
r_j	Integer	A random number generated by the KGC to compute a private key of user j, for $j = A, B, C$
n	Integer	The order of points on the curve
(D_j, s_j)	A point on the curve, integer	User j's private key as defined in [14, 16], for $j = A, B, C$
P_i	A point on the curve	A base point on the elliptic curve
Q	A point on the curve	The user's public key defined in the ECDSA algorithm
P_{pub_i}	A point on the curve	The public key of KGC_i
K_A^B, K_B^A	A point on the curve	The shared key between users A and B
s_j	Integer	The private key of user j defined in our proposed system, for $j = A, B, C, X$
(ID_j, R_j)	A pair of strings, a string representing a point on the curve	User j's public key defined in our proposed system, for $j = A, B, C, X$
(r, z)	A pair of integers	Digital signature (r, z) for message M
x_j	Integer	Session private key of user j, where $j = A, B, C, X$

the user's ID hashed using hash function $HF_1 : \{0,1\}^* \rightarrow Z_n^*$. Note that P is a base point on the elliptic curve of order n.

The signature signing and verification process in this work follow the ECDSA algorithm [14]. In the signing process, s_A is used to sign the message from user A, and then $(D_A, r(ECDSA), s(ECDSA))$ is sent out as his/her message signature. $r(ECDSA)$ is a random number defined in ECDSA and $s(ECDSA)$ is the digital signature computed in ECDSA. To verify the signature, D_A, h, P_{pub} are used by the receiver to compute $Q = D_A + h \cdot P_{pub}$, and then Q is used as user A's public key in the ECDSA algorithm.

The advantages of the approach in [14] are that the authors implemented a system that relies on ECDSA; it has been unarguably proved that ECDSA offers strong security with efficient performance. Despite this, the main disadvantage of their research lies in using part of the private key in the algorithm. D_A is part of the private key of user A, so D_A is secret for user A only, albeit that it is sent out to prove the authenticity of user A's signature. Therefore, the private key of a user is not confidential, which conflicts with the concept of a public-key cryptosystem.

3.2 ID-based Group Key Agreement without Pairing [16]

In this key agreement protocol, the KGC generates private key $sk_1 \in Z_p^*$ and public key $P_{pub} = sk_1 \cdot P$, where P is a base point on the elliptic curve. Random number r_A generated by the KGC is used to compute the private key for user A. On the user side, the public key for user A is the user's ID (e.g., name, phone number, etc.). The private key for user A is (s_A, D_A); D_A and s_A are respectively computed by the KGC as $D_A = r_A \cdot P$ and $s_A = (r_A + sk_1 \cdot h_A) \bmod p$, where $h_A = HF_2(ID_A)$ is the user's ID hashed using hash function $HF_2 : \{0,1\}^* \rightarrow \{0,1\}^k$, for which k is the bit length of prime p. Similarly, the KGC randomly produces r_B to generate $D_B = r_B \cdot P$ and $s_B = (r_B + sk_1 \cdot h_B) \bmod p$, where $h_B = HF_2(ID_B)$ for user B.

The simplified version of the key agreement algorithm in [16] between two users is as follows. Session private key $x_A \in Z_p^*$ is randomly chosen for user A and $T_A = x_A \cdot P$ is computed. T_A, D_A is sent on behalf of user A to user B and user A receives T_B, D_B from user B, where $T_B = x_B \cdot P$. Following this, K_A^B is computed for user A as follows:

$$\begin{aligned} K_A^B &= (s_A T_B + x_A(D_B + HF_2(ID_B) \cdot P_{pub})) \\ &= ((r_A + sk_1 \cdot HF_2(ID_A)) \cdot x_B P \\ &\quad + x_A(r_B \cdot P + HF_2(ID_B) sk_1 \cdot P)) \\ &= ((r_A P + sk_1 \cdot P \cdot HF_2(ID_A)) x_B \\ &\quad + x_A \cdot P(r_B + HF_2(ID_B) \cdot sk_1)) \\ &= ((D_A + HF_2(ID_A) \cdot P_{pub}) x_B + T_A s_B) \\ &= (s_B T_A + x_B(D_A + HF_2(ID_A) \cdot P_{pub})). \end{aligned}$$

Similarly, $K_B^A = (s_B T_A + x_B(D_A + HF_2(ID_A) \cdot P_{pub}))$ is computed for user B. As a result, the agreed key for messages between users A and B is $K_A^B = K_B^A$.

The disadvantage of the approach in [16] is the same as that in [14]; i.e., part of user j's private key D_j must be sent to proceed with the key agreement algorithm. Thus, the private key is no longer secret, and that violates the concept of a public-key cryptosystem. Moreover, this research is limited to users that belong to the same KGC, while in real-life situations, communication is usually between users from different KGCs.

4 The Proposed Pairing-free IBC

This is based on ECC and consists of encryption, digital signature, and key exchange schemes that make use of the same initial definition and key extraction parts. To demonstrate how this cryptosystem works, let us assume that there are three parties: users A, B, and C. Users A and B belong to KGC_1, while user C belongs to KGC_2. Their public and private keys are defined as follows:

For the rest of the paper, we define G as the cyclic additive group of points on an elliptic curve over finite field E/F_p.

4.1 System Parameter Definition

KGC_1 defines base point $P_1 \in G$ on the elliptic curve in which the order is a large value n and $nP_1 = O$, a hash function $HF_3 : \{0,1\}^* \rightarrow Z_n^*$, and master key $sk_1 \in [1, n-1]$. Subsequently, KGC_1 computes its public key $P_{pub_1} = (sk_1 \cdot P_1)$ and broadcasts $< G, P_1, P_{pub_1}, HF_3 >$ as the system parameters of the KGC_1 domain. In the same way, KGC_2 defines base point $P_2 \in G$, master key $sk_2 \in [1, n-1]$, and hash function $HF_3 : \{0,1\}^* \rightarrow Z_n^*$ and computes the corresponding $P_{pub_2} = (sk_2 \cdot P_2)$. KGC_2 keeps sk_2 secret and broadcasts $< G, P_2, P_{pub_2}, HF_3 >$ as the system parameters of the KGC_2 domain.

4.2 Key Extraction

In the key extraction phase, the users' public and private keys and some parameters used in the cryptosystem are defined. c_1 and c_2 are strings that are concatenated with the user's ID to generate the private key, which offers better security than the other algorithms. The procedure for defining the key is as follows.

4.2.1 Key Extraction for User A

KGC_1 chooses random number $r_A \in [1, n-1]$ to compute private key s_A for user A as $s_A = (sk_1 \cdot HF_3(ID_A) + r_A \cdot HF_3(ID_A||c_1)) \bmod n$. KGC_1 then computes $R_A = r_A \cdot P_1$ and sends s_A, R_A to user A through a secure channel. Upon receiving the message, s_A is kept secret for user A and string (ID_A, R_A) is announced as his/her public key. Note that R_A is composed of coordinate (x, y).

4.2.2 Key Extraction for User B

KGC_1 chooses random number $r_B \in [1, n-1]$ to compute private key s_B for user B as $s_B = (sk_1 \cdot HF_3(ID_B) + r_B \cdot HF_3(ID_B||c_1)) \bmod n$. KGC_1 then computes $R_B = r_B \cdot P_1$ and sends s_B, R_B to user B securely. Upon receiving the message, s_B is kept secret for user B and string (ID_B, R_B) is announced as his/her public key.

The above key definition has two advantages. First, it can prevent situations where two or more users cooperate to determine the KGC's master key. Second, if a user's private key is compromised, the KGC can easily generate a new secret key for that user. For example, if s_A is compromised, the KGC can randomly generate a new r_A to recompute the new private key.

4.2.3 Key Extraction for User C

The private key of user C, who is a member of KGC_2, can be computed as $s_C = (sk_2 \cdot HF_3(ID_C) + r_C \cdot HF_3(ID_C||c_2)) \bmod n$. Thus, the corresponding public key is (ID_C, R_C), where $R_c = r_c \cdot P_2$.

The pairing-free IBE, digital signature, and key exchange algorithms based on the above definition are introduced in the following subsections.

4.3 The Encryption and Decryption Scheme

Using the system parameters in the key extraction phase, encryption and decryption in the pairing-free ID-based key system can be achieved as follows: Let $HF_4 : \{(x, y)\} \rightarrow (0,1)^m$.

4.3.1 Encryption

When user B wants to send a secure message to user A, the encrypted message is computed by using user A's public key as follows. For user B, $x_B \in Z_n^*$ is randomly generated and $EncM = M \oplus HF_4((P_{pub_1} \cdot HF_3(ID_A) + R_A \cdot HF_3(ID_A||c_1)) \cdot x_B)$ is computed. Subsequently, encrypted message $((x_B \cdot P_1), EncM)$ is sent to user A.

4.3.2 Decryption

Message EncM is decrypted for user A by computing $DecM = EncM \oplus HF_4(s_A \cdot (x_B \cdot P_1))$.

4.3.3 Proof of the Correctness of the Scheme

The proof is as follows:

$$\begin{aligned} DecM &= EncM \oplus HF_4(s_A \cdot (x_B \cdot P_1)) \\ &= EncM \oplus HF_4((sk_1 \cdot HF_3(ID_A) \\ &\quad + r_A \cdot HF_3(ID_A||c_1)) \cdot (x_B \cdot P_1)) \\ &= EncM \oplus HF_4((sk_1 P_1 \cdot HF_3(ID_A) \\ &\quad + r_A P_1 \cdot HF_3(ID_A||c_1)) \cdot x_B) \\ &= EncM \oplus HF_4((P_{pub_1} \cdot HF_3(ID_A) \\ &\quad + R_A \cdot HF_3(ID_A||c_1)) \cdot x_B) \\ &= M. \end{aligned}$$

4.4 The ID-based Digital Signature Scheme

Our ID-based digital signature algorithm is based on ECDSA, the details of which can be found in [1]. In the algorithm, let HF_5 be cryptographic hash function SHA-1 giving a 160-bit integer value as the output. When user A wants to send a message together with a signature to user B, signature pair (r, z) is generated for message M by computing the following steps:

1) Randomly generate integer $k \in [1, n-1]$ for user A, and then compute point $(x, y) = k \cdot P_1$.

2) Computes the first portion of signature $r = x \bmod n$, for $r \neq 0$.

3) Hash message M using $e = HF_5(M)$.

4) Compute the second portion of signature $z = k^{-1}(e + s_A r) \bmod n$, for $z \neq 0$.

5) Send (r, z) as the signature of message M to user B.

When user B receives (r, z) from user A, the following steps are executed to verify the message:

1) Compute $e = HF_5(M)$;

2) Compute

$$\begin{aligned} Q &= P_{pub_1} \cdot HF_3(ID_A) + R_A \cdot HF_3(ID_A||c_1) \\ &= sk_1 \cdot P_1 \cdot HF_3(ID_A) + r_A \cdot P_1 \cdot HF_3(ID_A||c_1) \\ &= (sk_1 \cdot HF_3(ID_A) + r_A \cdot HF_3(ID_A||c_1)) \cdot P_1 \\ &= s_A \cdot P_1 \end{aligned}$$

3) Compute $w = z^{-1} \bmod n$;

4) Compute $u_1 = ew$ and $u_2 = rw$;

5) Compute point $(x_1, y_1) = u_1 \cdot P_1 + u_2 \cdot Q$;

6) Compute $v = x_1 \bmod n$;

7) Accept the signature (r, z) from user A if and only if $v = r$.

In ECDSA, it has been proved that if Q is the corresponding public key of the private key used in the signing process, then $v = r$. As we have followed the ECDSA process and because $Q = s_A P_1$, i.e., Q is equivalent to user A's public key in ECDSA, we can conclude that the proposed algorithm is valid.

4.5 The Key Exchange Scheme

For practical implementation, the proposed algorithm is designed to implement key exchange between parties from different KGCs. In the following example, we demonstrate key exchange between user A in KGC_1 and user C in KGC_2. The process begins with parameter generation and exchange between the parties.

1) $x_A \in [1, n-1]$ is randomly chosen for user A and then $(x_A \cdot P_2)$ is sent to user C.

2) $x_C \in [1, n-1]$ is chosen for user C and then $(x_C \cdot P_1)$ is sent to user A, after which both systems cooperate to produce a shared key for the users.

3) K_A^C is computed for user A: $K_A^C = s_A(x_C \cdot P_1) + x_A(R_C \cdot HF_3(ID_C||c_2) + P_{pub_2} \cdot HF_3(ID_C))$.

4) K_C^A is computed for user C: $K_C^A = (P_{pub_1} \cdot HF_3(ID_A) + R_A \cdot HF_3(ID_A||c_1))x_C + (x_A \cdot P_2)s_C$, after which the shared key between users A and C becomes $K_A^C = K_C^A$.

The following proof demonstrates that K_A^C and K_C^A are equal. Recall that

$$\begin{aligned} s_A &= (sk_1 \cdot HF_3(ID_A) + r_A \cdot HF_3(ID_A||c_1)) \bmod n, \\ s_C &= (sk_2 \cdot HF_3(ID_C) + r_C \cdot HF_3(ID_C||c_2)) \bmod n, \\ R_A &= r_A \cdot P_1, \\ R_C &= r_C \cdot P_2, \\ P_{pub_1} &= (sk_1 \cdot P_1), \\ P_{pub_2} &= (sk_2 \cdot P_2). \\ K_A^C &= s_A(x_C \cdot P_1) + x_A(R_C \cdot HF_3(ID_C||c_2) \\ &\quad + P_{pub_2} \cdot HF_3(ID_C)) \\ &= (sk_1 \cdot HF_3(ID_A) + r_A \cdot HF_3(ID_A||c_1)) \cdot (x_C \cdot P_1) \\ &\quad + x_A((r_C \cdot P_2) \cdot HF_3(ID_C||c_2) \\ &\quad + (sk_2 \cdot P_2) \cdot HF_3(ID_C)) \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_A) + r_A \cdot P_1 \cdot HF_3(ID_A||c_1)) \cdot x_C \\ &\quad + x_A \cdot P_2(r_C \cdot HF_3(ID_C||c_2) + sk_2 \cdot HF_3(ID_C)) \\ &= (P_{pub_1} \cdot HF_3(ID_A) + R_A \cdot HF_3(ID_A||c_1))x_C \\ &\quad + (x_A \cdot P_2)s_C \\ &= K_C^A. \end{aligned}$$

As illustrated above, the proposed encryption algorithm, digital signature, and key exchange processes make use of the same key definition and system parameters. The other main advantage of the key definition is that if the current user's private key is compromised, we can change the private key easily without affecting the user's ID. However, since R_A is not authenticated, there is the chance that an attacker will try to change the value of R_A to carry out attacks such as man-in-the-middle or using a fake signature. Fortunately, our proposed system can endure such attacks, as discussed in the next section.

5 Security and Performance Analysis of the Proposed Algorithm

In this section, we discuss the security of our proposed cryptosystem. The security analysis was conducted using a man-in-the-middle attack on the encryption and key exchange schemes. Moreover, problems with private key recovery and fake signatures for the digital signature scheme are discussed. In the last subsection, we analyze and compare the performance between our proposed pairing-free scheme with some well-known pairing-based ones.

5.1 Security Analysis of the Encryption Algorithm

One major issue for an encryption system is vulnerability to a man-in-the-middle attack. In brief, this is a situation

whereby an eavesdropper tries to intercept messages between users and sometimes modifies them without being detected. In this analysis, we suppose that user X, who is a member of KGC_1, impersonates user A and modifies user A's public key $ID_A.R_A$ to fraudulently created public key $ID_A.R_X$. Let us define user X's private keys as $s_X = (sk_1 \cdot HF_3(ID_X) + r_X \cdot HF_3(ID_X||c_1)) \bmod n$. The fake public key of user $A = ID_A.R_X$, where $R_X = r_X \cdot P_1$. Later, user B wants to send a secret message to user A. However, the message from user B is encrypted with the fraudulent public key, $ID_A.R_X$. Thus, message M is encrypted by computing: Random $x_B \in Z_n^*$ and $EncM = M \oplus HF_4((P_{pub_1} \cdot HF_3(ID_A) + R_X \cdot HF_3(ID_A||c_1)) \cdot x_B)$. After that, user B sends $((x_B \cdot P_1), EncM)$ to user A. User X eavesdrops on message $((x_B \cdot P_1), EncM)$ in the network system and he/she tries to recover the message M by using his/her private key s_X. However, user X fails to recover message M from EncM as we demonstrate below:

$$\begin{aligned} DecM &= EncM \oplus HF_4(s_X \cdot (x_B \cdot P_1)) \\ &= EncM \oplus HF_4(s_X \cdot (x_B \cdot P_1)) \\ &= EncM \oplus HF_4((sk_1 \cdot HF_3(ID_X) \\ &\quad + r_X \cdot HF_3(ID_X||c_1)) \cdot (x_B \cdot P_1)) \\ &= EncM \oplus HF_4((sk_1 P_1 \cdot HF_3(ID_X) \\ &\quad + r_X \cdot P_1 \cdot HF_3(ID_X||c_1)) \cdot x_B) \\ &= M \oplus HF_4((P_{pub_1} \cdot HF_3(ID_A) \\ &\quad + R_X \cdot HF_3(ID_A||c_1)) \cdot x_B) \\ &\quad \oplus HF_4((P_{pub_1} \cdot HF_3(ID_X) \\ &\quad + R_X \cdot HF_3(ID_X||c_1)) \cdot x_B) \\ &\neq M. \end{aligned}$$

Thus, the probability that user X can carry out a man-in-the-middle attack is $prob\{HF_3(ID_A) = HF_3(ID_X)\} \times prob\{HF_3(ID_A||c_1) = HF_3(ID_X||c_1)\}$. One interesting question is how to tell whether public R_A is genuine. We can test the genuineness of R_A by using the protocol; i.e., we can encrypt a message with user A's public key and send to user A. If user A responds with a correct message, we can tell that public R_A is genuine.

5.2 Security Analysis of the Digital Signature Scheme

The main problem with digital signatures is in situations where user X impersonates user A and sends a message with a fake digital signature to a receiver. If the receiver verifies the fake signature by using the fake public key of user A, i.e. $ID_A.R_X$, then the attack is successful. However, our proposed digital scheme can resist such attack as described below.

User X creates a fake digital signature for message M as follows:

1) Randomly create integer $k \in [1, n-1]$, and then compute $(x, y) = k \cdot P_1$;
2) Compute first signature $r = x \bmod n$, for $r \neq 0$;
3) Hash message M by using $e = HF_5(M)$;
4) Compute $z = k^{-1}(e + s_X r) \bmod n$, for $z \neq 0$;
5) Send (r, z) to user B.

Upon receiving the fake signature, user signature (r, z) is verified for user B by using the fake public key $(ID_A.R_X)$ of user A. The signature verification process is shown below.

1) Compute $e = HF_5(M)$;
2) Compute $Q = ((P_{pub_1} \cdot HF_3(ID_A)) + (R_X \cdot HF_3(ID_A||c_1)))$;
3) Compute $w = z^{-1} \bmod n$;
4) Compute $u_1 = ew$ and $u_2 = rw$;
5) Compute point $(x_1, y_1) = u_1 \cdot P_1 + u_2 \cdot Q$;
6) Compute $v = x_1 \bmod n$;
7) Compare v with r.

According to ECDSA, if $r = v$, then the verification is successful. As we showed in Section 4, $r = v$ if $Q = s_X \cdot P_1$. However, the above calculation shows that $Q = ((P_{pub_1} \cdot HF_3(ID_A)) + (R_X \cdot HF_3(ID_A||c_1))) \neq s_X \cdot P_1$. Therefore, (r, z) is rejected on behalf of user B as a fake signature. Furthermore, the probability of an attack being successful is $prob\{HF_3(ID_A) = HF_3(ID_X)\} \times prob\{HF_3(ID_A||c_1) = HF_3(ID_X||c_1)\}$. Another benefit that our proposed protocol inherits from ECDSA is that ECDSA is secure and can prevent private key recovery from signature (r, z). As the calculation of pair (r, z) in our scheme follows ECDSA, the scheme can endure a key recovery attack.

5.3 Security Analysis of the Key Exchange Scheme

A man-in-the-middle-attack is a major concern for a key exchange algorithm. In the key exchange process, when the parameters for two users (e.g., users A and B) are exchange when proceeding with the key exchange process, user X (the attacker) can impersonate user B and take part in the key exchange process with user A. The system interprets that an agreed key between users A and B has been created, but it is really between users A and X. In the same way, user X impersonates user A and takes part in the key exchange process with user B. Hence, user X can eavesdrop on or edit the communication between users A and B.

In this subsection, the security for two circumstances of using the key exchange algorithm is analyzed: two parties belonging to different KGCs and two in the same KGC.

5.3.1 Two Parties on Different KGCs

Let us assume that a shared key between user A belonging to KGC_1 and user C belonging to KGC_2 needs to be established. Meanwhile, the attacker (user X) belongs to KGC_1. In the key exchange process, the algorithm described in Section 4.5 is executed to establish a shared key for user A. Afterward, $(x_A \cdot P_2)$ on behalf of user A is sent out into the network. At that time, user X intercepts the message and sends $(x_X \cdot P_1)$ to user A. Upon receiving the message, the fake public key of user C, $(ID_C.R_X)$, is used by the system to compute a shared key for user A as follows:

$$\begin{aligned} K_A^X &= s_A \cdot (x_X \cdot P_1) + x_A(R_X \cdot HF_3(ID_C||c_2) \\ &\quad + P_{pub_2} \cdot HF_3(ID_C)) \\ &= (sk_1 \cdot HF_3(ID_A) + r_A \cdot HF_3(ID_A||c_1)) \\ &\quad \cdot (x_X \cdot P_1) + x_A((r_X \cdot P_2) \cdot HF_3(ID_C||c_2) \\ &\quad + (sk_2 \cdot P_2) \cdot HF_3(ID_C)) \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_A) + r_A \cdot P_1 \cdot HF_3(ID_A||c_1)) \\ &\quad \cdot x_X + x_A \cdot P_2(r_X \cdot HF_3(ID_C||c_2) \\ &\quad + sk_2 \cdot HF_3(ID_C)). \end{aligned}$$

Meanwhile, $(x_A \cdot P_2)$ is used by user X to compute

$$\begin{aligned} K_X^A &= (P_{pub_1} \cdot HF_3(ID_A) + R_A \cdot HF_3(ID_A||c_1))x_X \\ &\quad + (x_A \cdot P_2)s_X \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_A) + r_A \cdot P_1 \cdot HF_3(ID_A||c_1)) \\ &\quad \cdot x_X + x_A \cdot P_2(r_X \cdot HF_3(ID_X||c_2) \\ &\quad + sk_2 \cdot HF_3(ID_X)). \end{aligned}$$

Consequently, we can see that $K_A^X \neq K_X^A$, and thus user X fails to establish a shared key with user A. Moreover, the probability that the attack is successful is $prob\{HF_3(ID_C||c_2) = HF_3(ID_X||c_2)\} \times prob\{HF_3(ID_C) = HF_3(ID_X)\}$.

In the same way, user X impersonates user A and tries to establish a shared key with user C. In this process, $(x_C \cdot P_1)$ is sent to the network on behalf of user C, and user X intercepts the message and sends $(x_X \cdot P_2)$ back to user C. Next, the fake public key of user A $(ID_A.R_X)$ is used to compute a shared key for user C as follows:

$$\begin{aligned} K_C^X &= (P_{pub_1} \cdot HF_3(ID_A) + R_X \cdot HF_3(ID_A||c_1))x_C \\ &\quad + (x_X \cdot P_2) \cdot s_C \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_A) \\ &\quad + r_X \cdot P_1 \cdot HF_3(ID_A||c_1)) \cdot x_C \\ &\quad + x_X \cdot P_2(r_C \cdot HF_3(ID_C||c_2) \\ &\quad + sk_2 \cdot HF_3(ID_C)). \end{aligned}$$

Meanwhile, user X computes K_X^C using $(x_C \cdot P_1)$:

$$\begin{aligned} K_X^C &= s_X(x_C \cdot P_1) + x_X(R_C \cdot HF_3(ID_C||c_2) \\ &\quad + P_{pub_2} \cdot HF_3(ID_C)) \\ &= (sk_1 \cdot HF_3(ID_X) + r_X \cdot HF_3(ID_X||c_1)) \\ &\quad \cdot (x_C \cdot P_1) + x_X((r_C \cdot P_2) \cdot HF_3(ID_C||c_2) \\ &\quad + (sk_2 \cdot P_2) \cdot HF_3(ID_C)) \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_X) \\ &\quad + r_X \cdot P_1 \cdot HF_3(ID_X||c_1)) \cdot x_C \\ &\quad + x_X \cdot P_2(r_C \cdot HF_3(ID_C||c_2) \\ &\quad + sk_2 \cdot HF_3(ID_C)). \end{aligned}$$

Hence, we can see that $K_C^X \neq K_X^C$, and thus user X fails to establish a shared key with user C. Moreover, the probability that user X can establish a shared key with user C is $prob\{HF_3(ID_A) = HF_3(ID_X)\} \times prob\{HF_3(ID_A||c_1) = HF_3(ID_X||c_1)\}$. Even if user X eavesdrops on the communication between users A and C, he/she cannot recover or edit the message. We can conclude that the proposed scheme can prevent a man-in-the-middle attack in cases where the key exchange parties are members of different KGCs.

5.3.2 Two Parties on the Same KGC

Let us assume that users A, B, and X belong to KGC_1. In the key exchange process, $(x_A \cdot P_1)$ is sent out into the network on behalf of user A, and then user X intercepts the message and sends $(x_X \cdot P_1)$ to user A. The fake public key of user B $(ID_B.R_X)$ is used on behalf of user A to compute a shared key as follows:

$$\begin{aligned} K_A^X &= s_A \cdot (x_X \cdot P_1) + x_A(R_X \cdot HF_3(ID_B||c_1) \\ &\quad + P_{pub} \cdot HF_3(ID_B)) \\ &= (sk_1 \cdot HF_3(ID_A) + r_A \cdot HF_3(ID_A||c_1)) \cdot (x_X \cdot P_1) \\ &\quad + x_A((r_X \cdot P_1) \cdot HF_3(ID_B||c_1) \\ &\quad + (sk_1 \cdot P_1) \cdot HF_3(ID_B)) \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_A) + r_A \cdot P_1 \cdot HF_3(ID_A||c_1)) \cdot x_X \\ &\quad + x_A \cdot P_1(r_X \cdot HF_3(ID_B||c_1) + sk_1 \cdot HF_3(ID_B)). \end{aligned}$$

Meanwhile, user X uses $(x_A \cdot P_1)$ to compute a shared key:

$$\begin{aligned} K_X^A &= (P_{pub} \cdot HF_3(ID_A) + R_A \cdot HF_3(ID_A||c_1))x_X \\ &\quad + (x_A \cdot P_1) \cdot s_X \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_A) + r_A \cdot P_1 \cdot HF_3(ID_A||c_1)) \cdot x_X \\ &\quad + x_A \cdot P_1(r_X \cdot HF_3(ID_X||c_1) + sk_1 \cdot HF_3(ID_X)). \end{aligned}$$

According to the above computation, we can see that $K_A^X \neq K_X^A$. Thus, user X fails to establish a shared key with user A. Furthermore, the probability that the attack is successful is $prob\{HF_3(ID_B||c_1) = HF_3(ID_X||c_1)\} \times prob\{HF_3(ID_B) = HF_3(ID_X)\}$. In the same way, user X impersonates user A and tries to establish a shared key with user B. $(x_B \cdot P_1)$ is sent out into the network on behalf of user B, and then user X intercepts the message

and sends $(x_X \cdot P_1)$ to user B. The fake public key of user A $(ID_A.R_X)$ is then used to compute the shared key for user B as

$$\begin{aligned} K_B^X &= (P_{pub} \cdot HF_3(ID_A) + R_X \cdot HF_3(ID_A||c_1))x_B \\ &\quad +(x_X \cdot P_1)s_B \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_A) \\ &\quad +r_X \cdot P_1 \cdot HF_3(ID_A||c_1)) \cdot x_B \\ &\quad +x_X \cdot P_1(r_B \cdot HF_3(ID_B||c_1) \\ &\quad +sk_1 \cdot HF_3(ID_B)). \end{aligned}$$

Meanwhile, user X uses $(x_B \cdot P_1)$ to compute K_X^B as follows:

$$\begin{aligned} K_X^B &= s_X(x_B \cdot P_1) + x_X(R_B \cdot HF_3(ID_B||c_1) \\ &\quad +P_{pub_1} \cdot HF_3(ID_B)) \\ &= (sk_1 \cdot HF_3(ID_X) \\ &\quad +r_X \cdot HF_3(ID_X||c_1)) \cdot x_B \cdot P_1 \\ &\quad +x_X((r_B \cdot P_2) \cdot HF_3(ID_B||c_1) \\ &\quad +(sk_1 \cdot P_1) \cdot HF_3(ID_B)) \\ &= (sk_1 \cdot P_1 \cdot HF_3(ID_X) \\ &\quad +r_X \cdot P_1 \cdot HF_3(ID_X||c_1)) \cdot x_B \\ &\quad +x_X \cdot P_1(r_B \cdot HF_3(ID_B||c_1) \\ &\quad +sk_1 \cdot HF_3(ID_B)). \end{aligned}$$

Hence, we can see that $K_B^X \neq K_X^B$, and so user X fails to establish a shared key with user B. Moreover, the probability that the attack is successful is $prob\{HF_3(ID_A) = HF_3(ID_X)\} \times prob\{HF_3(ID_A||c_1) = HF_3(ID_X||c_1)\}$. From the findings of the above analysis, it can be seen that our proposed algorithm can prevent man-in-the-middle attacks if users are members of the same KGC or otherwise.

5.4 Performance Analysis of the Proposed Cryptosystem

Table 3 reports the results of a performance comparison between our proposed pairing-free scheme with well-known pairing-based ones.

Table 4 provides the results of a performance comparison between the proposed pairing-free scheme with other well-known pairing-free ones.

6 Conclusions

A pairing-free IBC using ECC and consisting of IBE, digital signature, and key exchange schemes was presented. All of the schemes use the same public and private key definitions, which makes the implementation of the system easy. The main advantage of the proposed key definition system is that if the user's private key is compromised, the KGC can easily generate a new one. As for the ID-based key exchange scheme, the proposed system can cope with situations where the communicating parties are on different KGCs. This is useful for mobile network computing in real scenarios. Proof of its correctness and a security analysis were provided, and the durability of the proposed system to several types of attacks (man-in-the-middle and intercepting signatures) was established. The proposed pairing-free scheme was compared with some well-known pairing-based and pairing-free ones, the results of which show that our proposed scheme gave a better performance than the pairing-based ones.

References

[1] ANSI X9.62, "Public key cryptography for the financial services industry: The elliptic curve digital signature algorithm (ECDSA)," 1998.

[2] F. Benhamouda, G. Couteau, D. Pointcheval, and H. Wee, "Implicit zero-knowledge arguments and applications to the malicious setting," in *CRYPTO'15*, pp. 107-129, 2015.

[3] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 286-615, 2003.

[4] X. Cao, W. Kou, and X. Du, "A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges," *Information Sciences*, vol. 180, no. 15, pp. 2895-2903, 2010.

[5] C. Gentry and A. Silverberg, "Hierarchical id-based cryptography," in *ASIACRYPT'02*, LNCS, vol. 2501, pp. 548-566, 2002.

[6] J. Hoffstein, J. Pipher, and J. H. Silverman, *An Introduction to Mathematical Cryptography*, Springer, 2008.

[7] A. Joux, "A one round protocol for tripartite Diffie-Hellman," in *Algorithm Number Theory Symposium – ANTS-IV*, LNCS, vol. 1838, pp. 385-393, 2000.

[8] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Crypto'85*, pp. 47-53, 1985.

[9] H. Li, Y. Dai, and B. Yang, "Identity-based cryptography for cloud security," *IACR Cryptology*, pp. 169-169, vol. 2011, 2011.

[10] M. Kumar, C. Katti, and P. Saxena, "An identity-based blind signature approach for e-voting system," *I.J. Modern Education and Computer Science*, vol. 9, no. 10, pp. 47-54, 2017.

[11] S. Nathani, B. Tripathi, and S. Khatoon, "A dynamic id based authenticated group key agreement protocol from pairing," *International Journal of Network Security*, vol. 21, no. 4, pp. 582-591, 2019.

[12] Y. Ming, and H. Yuan, "Fully secure anonymous identity based broadcast encryption with group of prime order," *International Journal of Network Security*, vol. 21, no. 1, pp. 7-16, 2019.

[13] N. Smart, "An identity based authenticated key agreement protocol based on the weil pairing," *Electronics Letters*, vol. 38, no. 13, pp. 630-632, 2002.

Table 3: Comparison of well-known pairing-based schemes and the proposed pairing-free scheme

Algorithm	Scheme	Number of Pairings $\hat{e}(U,V)$	Number of Scalar Multiplications on the Elliptic Curve	Number of Integer Multiplications	Number of $\hat{e}(U,V)\cdot\hat{e}(U,V)$	Number of Pairings $\hat{e}(U,V)^a$
Key Extract	Pairing-based [3]	- -	1 i.e. $S_{ID}=sk_1\cdot Q_{ID}$	- -	- -	- -
	Pairing-free	- - -	1 i.e. $R_j=r_j\cdot P_i$	2 i.e. $sk_i\cdot HF_3(ID_j)$, $r_j\cdot HF_3(ID_j\|\|c_i)$	- - -	- - -
Encryption	Pairing-based [3]	- -	1 i.e. $r\cdot P$	- -	- -	1 i.e. $\hat{e}(P_{pub},Q_{ID})^r$
	Pairing-free	- - - -	4 i.e. $P_{pub_i}\cdot HF_3(ID_A)$, $(R_A\cdot HF_3(ID_A\|\|c_i))\cdot x_B$, $x_B\cdot P_i$	- - - -	- - - -	- - - -
Decryption	Pairing-based [3]	1 i.e. $\hat{e}(rP,S_{ID})$	-	-	-	-
	Pairing-free	- -	1 i.e. $s_A\cdot(x_B\cdot P_1)$	- -	- -	- -
Digital Signature Signing Process	Pairing-based [18]	- - - -	4 i.e. $R=k\cdot P$, $Z=k^{-1}(H_3(m)\cdot P$ $+H_4(R)\cdot S_{ID})$	- - - -	- - - -	- - - -
	Pairing-free	-	1 (i.e. $k\cdot P_i$)	2 (i.e. $k^{-1}(e+s_jr)$)	-	-
Digital Signature Verification Process	Pairing-based [18]	1 i.e. $\hat{e}(R,Z)$	-	-	1 i.e. $\hat{e}(P,P)^{H_3(m)}$ $\cdot\hat{e}(sk_1P,Q_{ID})^{H_4(R)}$	2 i.e. $\hat{e}(P,P)^{H_3(m)}$, $\hat{e}(sk_1P,Q_{ID})^{H_4(R)}$
	Pairing-free	-	4 i.e. $P_{pub_1}\cdot HF_3(ID_A)$, $R_A\cdot HF_3(ID_A\|\|c_1)$, $u_1\cdot P_1$, $u_2\cdot Q$	2 i.e. $u_1=ew$, $u_2=rw$	- - - - -	- - - - -
Key Exchange	Pairing-based [13]	2 i.e. $\hat{e}(aQ_B,P_{KGS})$, $\hat{e}(S_A,T_B)$	2 i.e. aQ_B, $T_A=aP$	-	1 i.e. $\hat{e}(aQ_B,P_{KGS})$ $\cdot\hat{e}(S_A,T_B)$	-
	Pairing-free	-	5 i.e. $s_A(x_C\cdot P_1)$, $x_A(R_C\cdot HF_3(ID_C\|\|c_2)$ $+P_{pub_2}\cdot HF_3(ID_C))$	- - - -	- - - -	- - - -

Table 4: Comparison of well-known pairing-free schemes and the proposed pairing-free scheme

Algorithm	Scheme	Number of Scalar Multiplications on the Elliptic Curve	Number of Integer Multiplications	Sending out the User's Private Key	Key Exchange between Users Belonging to Different KGCs
Key Extract	Jin *et al.* [14], Kumar &Tripathi [16]	1 i.e. $D_A = r_A \cdot P$	1 i.e. $s_A = (r_A + h \cdot sk_1)$	- -	-
	The proposed algorithm	1 i.e. $R_A = r_A \cdot P_1$	2 i.e. $s_A = sk_1 \cdot HF_3(ID_A) + r_A \cdot HF_3(ID_A\|\|c_1)$	- - -	- - -
Digital Signature Signing Process	Jin *et al.* [14]	1 i.e. $(x_1, y_1) = k \cdot P_1$	2 i.e. $s = k^{-1}(e + dr)$	Yes	-
	The proposed algorithm	1 i.e. $(x, y) = k \cdot P_1$	2 i.e. $z = k^{-1}(e + s_A r)$	No	-
Digital Signature Verification Process	Jin *et al.* [14]	3 i.e. $Q = D_A + h \cdot P_{pub}$, $u_1 \cdot P_1, u_2 \cdot Q$	2 i.e. $u_1 = ew$, $u_2 = rw$	- - -	- - -
	The proposed algorithm	4 i.e. $Q = P_{pub_1} \cdot HF_3(ID_A) + R_A \cdot HF_3(ID_A\|\|c_1)$, $u_1 \cdot P_1, u_2 \cdot Q$	2 i.e. $u_1 = ew$, $u_2 = rw$	- - -	- - -
Key Exchange	Kumar &Tripathi [16]	4 i.e. $T_A = x_A \cdot P$, $K_A^B = (s_A T_B + x_A(D_B + HF_2(ID_B) \cdot P_{pub}))$	- - -	Yes	No
	The proposed algorithm	5 i.e. $K_A^C = s_A(x_C \cdot P_1) + x_A(R_C \cdot HF_3(ID_C\|\|c_2) + P_{pub_2} \cdot HF_3(ID_C))$	- - -	No	Yes

[14] H. Jin, H. Debiao and C. Jianhua, "An identity based digital signature from ECDSA," in *Second International Workshop on Education Technology and Computer Science*, Wuhan, vol. 1, pp. 627-630, 2010.

[15] L. Lui, Z. Cao, W. Kong, and J. Wang, "On bilinear groups of a large composite order," *International Journal of Electronics and Information Engineering*, vol. 7, no. 1, pp. 1-9, 2017.

[16] A. Kumar and S. Tripathi, "Anonymous id-based group key agreement protocol without pairing," *International Journal of Network Security*, vol. 18, no. 2, pp. 263-273, 2016.

[17] C. Youngblood, "An introduction to identity-based cryptography," in *CSEP 590TU*, 2005.

[18] K. Paterson, "Id-based signatures from pairings on elliptic curves," *Electronics Letters*, vol. 38, no. 18, pp. 1025-1026, 2002

Biography

Poonsuk Ponpurmpoon received the B.A. degree in Information Studies from Ramkhamhaeng University, Thailand. He completed his M.Sc. and Ph.D. degrees in Computer Science and Information Systems from National Institute of Development Administration, Thailand. His research interests are in cryptography and information security.

Pipat Hiranvanichakorn received the B.E. degree in Electrical Engineering from Chulalongkorn University, Thailand, in 1977 and the M.E. and D.E. degrees in Information Processing from Tokyo Institute of Technology, Japan, in 1982 and 1985. He has been an associate professor of Computer Science at School of Applied Statistics, National Institute of Development Administration, Thailand. He is now an independent researcher at Suksoomboon Laboratory. His current research interests include natural language processing, computer networks, cryptography and information security.

Research on Network Anomaly Data Flow Intrusion Detection and Defense Under Self-Defending Network Architecture

Bing Bai

(Corresponding author: Bing Bai)

Shaanxi Police College

Xi'an, Shaanxi 710021, China

Email: tuibai36024@163.com

(Received Feb. 28, 2020; Revised and Accepted May 28, 2022; First Online June 23, 2022)

Abstract

A self-defending network (SDN) is easier to control and scale than traditional Internet architectures, but they also have to face malicious attacks, primarily Distributed Denial of Service (DDoS) attacks. This paper briefly introduced the basic architecture of SDN and DDoS attacks in SDN. Convolutional Neural Network (CNN) was combined with Long Short Term Memory (LSTM) for the detection and defense against DDoS attacks. The simulation experiments were conducted on the CNN+LSTM algorithm in the SDN built in the laboratory, and the CNN+LSTM algorithm was compared with K-means and CNN algorithms. The experimental results showed that the CNN+LSTM detection algorithm had higher recognition accuracy and efficiency; the SDN switch had a lower traffic share under the CNN+LSTM detection algorithm than the other two detection algorithms when facing DDoS attacks; as the number of DDoS attacks increased, the CNN+LSTM algorithm maintained stable recognition accuracy, recognition efficiency and switch traffic share.

Keywords: Convolutional Neural Network; Distributed Denial of Service; Long Short Term Memory; Self-defending Network

1 Introduction

The emergence of the Internet has greatly facilitated people's lives. Relying on the convenience of the Internet, people gradually upload information around them to the Internet, especially when using the Internet for online shopping, it is necessary to upload real address information to the Internet [15]. As the Internet becomes an indispensable part of people's daily life, if the Internet is paralyzed by malicious attacks, it will seriously affect the daily life related to the Internet [5], and the sensitive information stored on the Internet will be stolen by criminals, thus causing losses. Therefore, the Internet's ability to protect against anomalous attacks is crucial [9]. Saied *et al.* [11] proposed a specific feature (pattern)-based artificial neural network (ANN) algorithm to detect distributed denial of service (DDoS) attacks and found that the feature separated DDoS attack traffic from real traffic.

Yan *et al.* [14] proposed a multi-queue self-defending network (SDN) controller scheduling algorithm based on a time slice allocation policy. The algorithm took different time slice allocation policies according to the DDoS attack intensity and used SDN controllers to schedule processing flow requests from different switches to better protect protection of normal switches in the network under DDoS attacks. Simulation experiments verified the effectiveness of this algorithm. Sahay *et al.* [10] proposed an autonomous DDoS defense framework called ArOMA. The experimental results showed that ArOMA could effectively maintain the performance of video streams at a satisfactory level in the face of DDoS flooding attacks. This paper briefly introduced the basic architecture of SDN and DDoS attacks in SDN, combined Convolutional Neural Network (CNN) with Long Short Term Memory (LSTM) for the detection and protection against DDoS attacks, and simulated the CNN+LSTM algorithm in an SDN network established in a lab.

2 SDN Architecture

The basic structure of the Internet is mainly divided into core devices, links, and edge devices. As the scale of the Internet expands, the complexity of Internet control increases, and the traditional distributed Internet control model is divided into the Open System Interconnection (OSI) seven-layer model and the Transmission Control Protocol/Internet Protocol (TCP/IP) five-layer model, but the traditional Internet control model is self-

contained [3]. As the amount of data within the Internet increases and the functional structure changes, various complex network protocols are placed into the router. On the one hand, it causes a large burden to router memory; on the other hand, the complex protocols and routers that are not open to the outside lead to different Internets between different operators, and it is also difficult for Internet-related researchers to perform practical validation of research results in real networks.

In order to improve the efficiency of the Internet, make maintenance of the Internet easier, and reduce maintenance costs, SDN architecture is put forward. Compared with the 5- and 7-layer structures of the Internet, SDN has a 3-layer structure, as shown in Figure 1. Because of fewer layers, SDN is relatively easier to upgrade and maintain the network functions [1]. SDN separates the control plane and data plane in the network from each other. The control plane centralizes the control of the network and is divided into the application layer, the control layer, and the basic data layer. The application layer is connected to the control layer through an Application Programming Interface (API). The control layer programs and controls the application software in the application layer. Application programs in the application layer build an abstract view of the network according to the information given by the control layer to identify and defend against malicious information in the network [6]. SDN controller is in the SDN control layer. In addition to providing network information to the application layer through API, the SDN controller also receives application command information from the application layer and passes the command information to the basic data layer through the OpenFlow protocol. The data layer contains infrastructure such as SDN switches. The devices in the data layer do not have the ability to control other devices but execute the commands given by the control layer or forward the commands to other devices. When forwarding commands or data, it is the flow table, not the destination address, that determines the forwarding path.

3 Abnormal Data Detection Algorithm

3.1 DDOS Attacks

SDN separates the data control function from the data forwarding function and uses the SDN controller to deploy network data in a unified manner, reducing management difficulties. If a new operator joins the network, there is no need to add a new communication protocol to the routers on the Internet, and instead a new interface in the SDN controller is extended. The new operator can upload the request to the SDN controller by following the communication protocol of the interface, and then the SDN controller will order the SDN switch to transmit the network data according to the request. In this way, the network capacity can be more open, and network managers can manage and dispatch the network more easily.

However, even if the structure of the Internet is changed, the number of malicious attacks to which the Internet will be subjected will not be reduced. DDoS is a common malicious attack [7]. The DDoS attack uses the OF protocol mechanism between the control and data layers to launch a DDoS attack on the SDN controller and generates a large number of Packet_In requests and flow entries through a large number of false IPs to consume the computing resources of the controller and switch to achieve the purpose of paralyzing the network.

3.2 Detection and Defense Against DDoS Attacks

In order to ensure the security of SDN, it is necessary to detect the data in the network and intercept the abnormal data, especially the DDoS attack which is more likely to harm SDN. The basic principle of detecting abnormal data in the network is to extract the features of the data to be detected and classify the data according to the features to determine whether the data are abnormal or not. The commonly used detection algorithms are entropy-based detection method, clustering-based detection method [8], neural network-based detection method, etc.

Since the amount of data flowing on the Internet is large and the data change pattern is closer to nonlinearity, this paper chooses a neural network that is better at mining hidden nonlinear laws to perform DDoS detection [2]. In this paper, CNN is combined with LSTM. Figure 2 shows the detection and defense process of DDoS attacks after the combination of CNN and LSTM.

1) The statistics of flow entries at the entry end are collected from the SDN controller with the cycle of T. Then, temporal features are extracted from the statistics of flow entries at the entry end of the controller within T cycle. The temporal feature used in this paper is the hit rate of flow entries at the switch port in the collection cycle.

2) The temporal features of flow entries are input into the convolution layer of CNN for convolution operation by convolution kernels to further extract the local features of the temporal features of flow entries. Local features calculated by convolution kernels can also be combined into the overall features. The formula for convolution calculation by convolution kernels [12] is:

$$x_j^i = f(\sum_{j \in M} x_i^{l-1} W_{ij}^l + b_j^l),$$

where x_j^l is the feature map obtained by convolution of convolution kernels; x_j^{l-1} is the feature output after the last convolution and pooling; W_{ij}^l is the weight parameter; b_j^l is the bias amount; M is the number of convolution kernels, and $f(\cdot)$ is

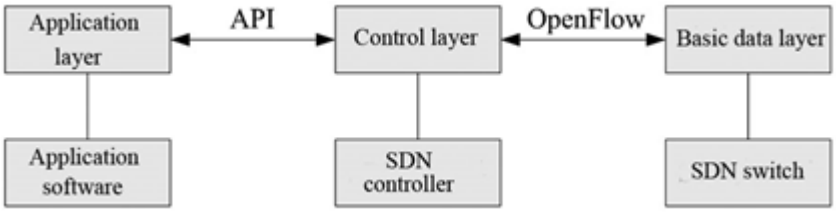


Figure 1: Basic network architecture of SDN

the activation function. The local feature map obtained by convolution is pooled and compressed in the pooling layer, thus reducing the amount of computation afterward. During compression, a pooling frame slides over the convolutional feature map, and the features within the pooling frame are averaged or maximized in every step to obtain the compressed feature map [13].

3) The convolution and pooling operations are repeated in CNN several times according to the demand. Then, the convolution features output from CNN are used as the inputs of LSTM. The input convolutional features are calculated and judged by using the input gate, forgetting gate and output gate in LSTM, and the corresponding calculation formulas are:

$$\begin{aligned} f_t &= \rho(b_f + U_f x_t + W_f h_{t-1}) \\ s_t &= f_t s_{t-1} + g_t \rho(b + U x_t + W h_{t-1}) \\ g_t &= \rho(b_g + U_g x_t + W_g h_{t-1}) \\ h_t &= \tan h(s_t) q_t \\ q_t &= \rho(b_q + U_q x_t + W_q h_{t-1}) \end{aligned}$$

where x_t is the t-th input sample, h_{t-1} and h_t are the hidden states of the $(t-1)$-th and t-th input samples, f_t is the output of the forgetting gate, b_f, U_f and W_f are the bias term and input term weight in the forgetting gate and the weight of the forgetting gate, s_t is the output of the loop gate, b, U and W are the bias term and input term weight in the loop gate and the weight of the loop gate, g_t is the external input gate unit, b_g, U_g and W_g are the bias term and input term weight in the input gate and the weight of the input gate, q_t is the output gate unit, b_q, U_q and W_q are the bias term and input term weight in the output gate and the weight of the output gate.

4) The data obtained from the output gate unit is classified and judged in the fully connected layer to determine whether the input sample is a DDoS attack or not. If it is a DDoS attack, the SDN switch port that collects that sample is blocked, and all the SDN switch ports are traversed to check whether there are blocked ports; if it is not a DDoS attack, the SDN switch is traversed directly to check whether there are blocked ports.

5) If no blocked port is found after traversing the SDN switch ports, it returns to Step 1; if there is a blocked port, the hit rate of the flow table in the blocked port is recorded.

6) If the flow table hit rate of the blocked port is greater than the flow table hit rate of the normal port, the port is not processed, and it returns to Step 1; if the flow table hit rate of the blocked port does not exceed the flow table hit rate of the normal port, the port is unblocked, and it returns to Step 1.

4 Simulation Experiments

4.1 Experimental Setup

The simulation experiments of DDoS attack detection and defense algorithms were conducted in a laboratory-built local area network, whose structure was SDN. A total of 16 servers were used, numbered 1 16. Server No. 1 was used as the SDN controller, servers No. 2 6 were used as the SDN switch, servers No. 7 and 8 were used as the servers for receiving packets, and servers No. 9 16 were used as the servers for sending packets. Server No. 1 was connected to servers No. 2 6 through the OF protocol. Server No. 2 was connected to servers No. 7 and 8. Server No. 3 was connected to No. 9 and 10. Server No. 4 was connected to No. 11 and 12,. Server No. 5 was connected to No. 13 and 14. Server No. 6 was connected to No. 15 and 16.

In order to verify the performance of the CNN+LSTM-based DDoS detection and protection algorithm, simulation experiments were conducted to test not only the CNN+LSTM algorithm but also the K-means-based detection algorithm and the CNN-based detection algorithm.

The principle of the K-means algorithm is as follows. Cluster centers are randomly selected from the data set according to the number of classes, and then the data features are classified to the nearest cluster center according to the Euclidean distance. The cluster center for every class of clusters is recalculated. Again, the data features are classified to the nearest cluster center according to the Euclidean distance. The above steps are repeated until convergence. The algorithm is unsupervised and depends mainly on the number of cluster centers, i.e., the number of classes. In this paper, the number of classes was set as 2, one was low traffic data and the other was non-low traffic data. The traffic was evaluated as abnormal when the low traffic data exceeded the set threshold, i.e., 30.

The relevant parameters of the CNN-based detection algorithm are as follows. It was composed of two convolution layers containing 32 convolution kernels in a size of 2 × 2, one pooling layer containing a 3 × 3 pooling frame, two convolution layers containing 16 convolution kernels in a size of 2 × 2, and one pooling layer containing a 3 × 3 pooling frame. The sigmoid activation function was used in the convolution layer [4]. Max-pooling was used in the pooling layer. The error was calculated by cross-entropy. The learning rate was set as 0.1.

The relevant parameters of the CNN+LSTM-based algorithm are as follows. The parameters of the convolution layer and pooling layer of CNN were consistent with those in the CNN algorithm. LSTM had two hidden layers. There were 1024 nodes in every hidden layer. The sigmoid activation function was used. Cross-entropy was also used during training. The learning rate was set as 0.1.

4.2 Experimental Methods

The DDoS intrusion detection data set contained SYN Flood attack, ACK Flood attack, UDP Flood attack, and ICMP Flood attack. When the DDoS attack was simulated, the normal traffic pattern was set as: the server sending packets had a 50% probability of sending packets per second to the server receiving packets; the switch connected to the server receiving packets selected the target server according to the weight of 7:3. The simulation was run in normal traffic mode for 10 s, and then the DDoS attack packet launched a 10 s attack targeting server No. 2 (SDN switch).

In order to verify the detection performance of the designed detection algorithms in the face of the traffic of different sizes, when a DDoS attack started, DDoS intrusion data were added to the original normal traffic. The recognition accuracy and average time consumption of the three detection algorithms were tested under 1000, 2000, 3000, 4000 and 5000 intrusion data, respectively. Moreover, the traffic share of server No. 2 in normal traffic mode and the traffic percentage in DDoS attack mode were recorded.

4.3 Experimental Results

Defending against DDoS attacks in SDN with the detection algorithm should not only consider the identification time but also ensure the accuracy of abnormal attack identification. If the recognition accuracy is low, it will lead to abnormal opening and closing of SDN ports, which will not only fail to guarantee security but also affect the normal users. In addition, the time consumption of the recognition algorithm will also affect the use of SDN. The faster the detection algorithm detects the data, the less likely it will affect the users. Figure 2 shows the recognition accuracy and average time consumption of three detection algorithms when facing DDoS attacks of different traffic sizes. First of all, the comparison of the recognition accuracy showed that under the same number of attacks, the CNN+LSTM-based detection algorithm had the highest accuracy, the CNN-based detection algorithm was the second, and the K-means-based detection algorithm was the lowest. The curve in Figure 2 showed that as the number of DDoS attacks increased, the recognition accuracy of both K-means-based and CNN-based detection algorithms decreased gradually, and the accuracy of the CNN+LSTM-based detection algorithm also decreased, but not significantly. Then, the average time consumed for identification was compared. Under the same number of attacks, the K-means-based detection algorithm consumed the longest time, followed by the CNN-based detection algorithm and the CNN+LSTM-based detection algorithm. The curve of the time consumption in Figure 2 also showed that as the number of DDoS attacks increased, the time consumed by K-means-based and CNN-based detection algorithms increased significantly, but the time consumed by the CNN+LSTM-based detection algorithm nearly had no change.

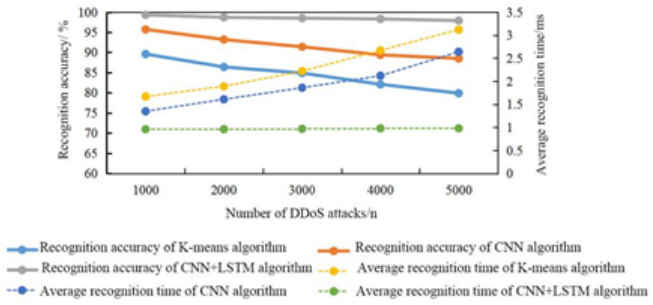


Figure 2: Identification performance and time consumption of three detection algorithms in the face of different numbers of DDoS attacks

Figure 3 shows the change in traffic share before and after the DDoS attack when server No. 2 used the three detection algorithms. It was seen from Figure 3 that in the normal traffic mode before the server was attacked, the traffic share of server No. 2 (SDN switch) was stable at about 50% regardless of the detection algorithm used by the server; while in the abnormal traffic mode under DDoS attack, the traffic share of the server under all the three detection algorithms increased, and the K-means detection algorithm increased the most, followed by the CNN-based algorithm and the CNN+LSTM-based algorithm. In addition, in the abnormal traffic mode, with the increase of DDoS attacks, the traffic share of the server that used K-means-based and CNN-based algorithms increased significantly, but that of the CNN+LSTM-based algorithm remained unchanged.

5 Conclusion

This paper briefly introduced the basic architecture of SDN and DDoS attacks in SDN and combined CNN with LSTM for the detection and defense of DDoS at-

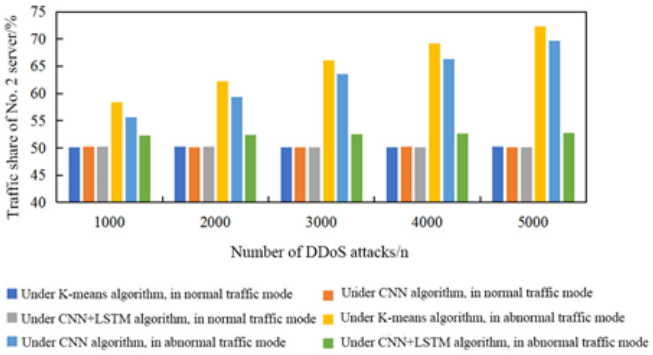


Figure 3: Server traffic share under the three detection algorithms in the face of different numbers of DDoS attacks

tacks. The simulation experiments were carried out on the CNN+LSTM detection algorithm in the SDN built in a laboratory. The CNN+LSTM-based algorithm was compared with K-means and CNN-based detection algorithms. The results are as follows. Under the same number of DDoS attacks, the CNN+LSTM-based detection algorithm had the highest recognition accuracy and recognition efficiency, the CNN-based detection algorithm was the second-highest, and the K-means-based detection algorithm was the lowest.

In addition, as the number of DDoS attacks increased, the recognition accuracy and recognition efficiency of K-means-based and CNN-based detection algorithms decreased, but the CNN+LSTM-based detection algorithm remained stable. In the normal traffic mode, the server traffic share was stable at 50% regardless of the detection algorithm; in the abnormal traffic mode, after the addition of DDoS attack data, the server traffic share increased, the K-means algorithm had the largest increase, and the CNN+LSTM-based algorithm had the smallest increase. In addition, as the number of DDoS attacks increased, the server traffic share increased, the server traffic share under K-means and CNN-based algorithms increased significantly, but the server traffic share under the CNN+LSTM algorithm remained stable.

References

[1] M. de Assis, A. H. Hamamoto, T. Abrao, M. L. Proença, "A game theoretical based system using holt-winters and genetic algorithm with fuzzy logic for DoS/DDoS mitigation on SDN networks," *IEEE Access*, vol. 5, pp. 9485-9496, 2017.

[2] S. Behal, J. Singh, "Detection and mitigation of DDoS attacks in SDN: A comprehensive review, research challenges and future directions," *Computer Science Review*, vol. 37, pp. 1-25, 2020.

[3] N. Hoque, D. K. Bhattacharyya, J. K. Kalita, "Botnet in DDoS attacks: Trends and challenges," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2242-2270, 2015.

[4] I. F. Kilwalaga, F. Sumadi, S. Syaifuddin, "SDN-honeypot integration for DDoS detection scheme using entropy," *KINETIK: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, vol. 5, no. 3, 2020.

[5] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80-84, 2017.

[6] H. Mahmood, D. Mahmood, Q. Shaheen, R. Akhtar, C. Wang, "S-DPS: An SDN-based DDoS protection system for smart grids," *Security and Communication Networks*, vol. 2021, pp. 1-19, 2021.

[7] R. Majid, R. Ujjan, Z. Pervez, K. Dahal, W. A. Khan, A. M. Khattak, B. Hayat, K. Kim, E. Kim, "Entropy based features distribution for anti-DDoS model in SDN," *Sustainability*, vol. 13, no. 3, 2021.

[8] M. P. Novaes, L. F. Carvalho, J. Lloret, M. L. Proença, "Adversarial deep learning approach detection and defense against DDoS attacks in SDN environments," *Future Generation Computer Systems*, vol. 125, no. 1, 2021.

[9] O. Osanaiye, K. K. R. Choo, M. Dlodlo, "Distributed denial of service (DDoS) resilience in cloud: Review and conceptual cloud DDoS mitigation framework," *Journal of Network & Computer Applications*, vol. 67, pp. 147-165, 2016.

[10] R. Sahay, G. Blanc, Z. Zhang, H. Debar, "ArOMA: an SDN based autonomic DDoS mitigation framework," *Computers & Security*, vol. 70, 2017.

[11] A. Saied, R. E. Overill, T. Radzik, "Detection of known and unknown DDoS attacks using artificial neural networks," *Neurocomputing*, vol. 172, pp. 385-393, 2016.

[12] M. P. Singh, A. Bhandari, "New-flow based DDoS attacks in SDN: Taxonomy, rationales, and research challenges - ScienceDirect," *Computer Communications*, vol. 154, pp. 509-527, 2020.

[13] A. Wani, S. Revathi, "DDoS detection and alleviation in IoT using SDN (SDIoT-DDoS-DA)," *Journal of The Institution of Engineers (India) Series B*, vol. 101, no. 3, pp. 117-128, 2020.

[14] Q. Yan, Q. Gong, F. R. Yu, "Effective software-defined networking controller scheduling method to mitigate DDoS attacks," *Electronics Letters*, vol. 53, no. 7, pp. 469-471, 2017.

[15] Q. Yan, R. Yu, Q. Gong, J. Li, "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 602-622, 2016.

Biography

Bing Bai, born in December 1985, has received the master's degree in computer technology from Xi'an University of technology in 2019. He is a lecturer and senior

engineer. His research interests are network security and network information technology.

An Improved User Identity Authentication Protocol for Multi-Gateway Wireless Sensor Networks

Liling Cao, Yu Zhang, Mei Liang, and Shouqi Cao
(Corresponding author: Shouqi Cao)

Department of Engineering Science and Technology, Shanghai Ocean University
Shanghai 201306, China
Email: sqcao@shou.edu.cn

(Received Apr. 16, 2021; Revised and Accepted Mar. 26, 2022; First Online May 17, 2022)

Abstract

As an important part of the IoT (IoT), wireless sensor networks (WSNs) have been widely studied and applied in many fields. However, wireless sensor networks are prone to various security risks due to their openness. This paper reviews the contents of Srinivas *et al.*'s protocol and finds that the protocol is vulnerable to stolen smart card attacks and cannot achieve time synchronization, forward secrecy, and user anonymity. Then an improved user identity authentication protocol IUIAP is proposed. The security of protocol IUIAP is demonstrated by security analysis using BAN logic, a formal analysis method proposed by Burrows, Abadi, and Needham. The performance comparison and efficiency analysis with other related protocols are carried out. The results show that the improved protocol IUIAP can solve the security defects of Srinivas *et al.*'s protocol and can provide various security properties with little more computation cost, such as resistance to stolen smart card attacks and insider attacks, anonymous protection for users, protection for known keys, protection for forwarding secrecy and without problems of time synchronization. Therefore, the improved protocol IUIAP is more secure and suitable for multi-gateway WSNs.

Keywords: Authentication; Multiple Gateways; Wireless Sensor Network

1 Introduction

The Internet of Things (IoT) is a huge network formed by extending and expanding the foundation of the Internet and combining various information sensing devices with the Internet. The Internet of Things has a wide range of applications, including smart homes, smart cities, public health, electronic medical care, energy management, etc. As an important part of the Internet of Things, wireless sensor networks(WSNs) have become the hottest issue in the research that combines wireless communication with computer networks [2, 17, 23, 25, 26].

The wireless sensor network is mainly composed of three parts, including gateway nodes, sensor nodes and users. The sensor nodes which are dispersedly distributed can collect the external environment information in real time. The gateway node is the most imperative part in WSN, which plays the role of connecting users and sensor nodes and receives and forwards the information to sensor nodes or users [11, 15]. The information transmission between gateway nodes, sensor nodes and users is public over an unprotected channel [13, 27]. The attackers can initiate active attacks to intercept, tamper, replay and forward the information, which causes that WSNs are vulnerable to multiple network attacks [1]. Fortunately, identity authentication technology has attracted extensive attention and been applied in WSNs to guarantee the security [24, 30]. However, WSNs is a flexible multihop self-organizing network on account of the emergence of sensor nodes leaving and joining and the change of sensors location [21]. In addition, resources in sensor nodes with small size and low cost are limited in computation, storage, and communication [18]. Therefore, it is a great challenge for researchers to propose a secure and efficient identity authentication protocol for multi-gateway WSNs to solve security problems of WSNs.

In 2009, Das *et al.* [5] proposed an efficient smart card-based password authentication protocol in wireless sensor networks for the first time, which provided basic guarantee for users to access real-time data in the sensor subsystem. Das *et al.* believed that their proposed protocol avoided many login users with the same login ID and stolen verifier attacks. In addition to denial of service attacks and node attacks, the proposed protocol could also resist other attacks in WSN.

In 2010, He *et al.* [9] and Nyang and Lee *et al.* [19] showed that Das's protocol was vulnerable to internal attacks and user impersonation attacks and failed to provide

anonymous protection. They proposed an enhanced two-factor user identity authentication scheme, which promoted user anonymity, prevented internal attacks and provided users with password change services. However, their scheme did not care about mutual authentication between all communication parties. Khan and Alghathbar *et al.* [14] devoted to improving Das's protocol and proposed an improved protocol that overcame the security loopholes mentioned above, including providing mutual authentication between all parties. Furthermore, Chen *et al.* [4] improved the confidentiality of the Das's protocol to ensure that legitimate users could use WSN in an insecure environment. But Yeh *et al.* [29] found that the improved method of Chen *et al.* provided a safe way to update users' passwords but was vulnerable to internal attacks. Meanwhile, Yeh *et al.* proposed a new user authentication protocol for WSN using smart cards and elliptic curve cryptography to avoid the security vulnerabilities of the Das *et al.*'s protocol. However, this solution did not resist internal attacks and provide user anonymity [24].

In 2013, Xue *et al.* [28] proposed a lightweight time credential-based mutual authentication and key agreement scheme, which was suitable for resource-constrained wireless sensor network environments, allowing legitimate users to access sensor data in any specific sensor nodes. Later, a lot of improvement schemes (see [8,10,12,16,22]) over Xue *et al.*'s scheme had been presented. Among them, Turkanovic *et al.* [22] used the concept of IoT in 2014 to propose a lightweight user authentication protocol based on hash function for heterogeneous Ad Hoc WSN. Farash *et al.* [7] pointed out some security weaknesses in the protocol of Turkanovic *et al.* and extended it to enhance its security.

In 2016, Amin *et al.* [1] found that Farash *et al.*'s scheme had some problems, including known session-specific temporary information attacks, offline password guessing attacks using stolen smart cards, attacks from the new smartcards, and user impersonation attacks. In addition, Farash *et al.*'s scheme could not achieve user anonymity and the key of the gateway node was also insecure. At the same time, Amin *et al.* proposed an efficient and powerful smart card-based user authentication and session key agreement protocol, which not only overcame the weakness of Farash *et al.*'s protocol, but also retained other security properties. However, Srinivas *et al.* [20] found that Amin *et al.*'s scheme did not resist many attacks like leakage of sensors secret keys and system key, server spoofing attack, user impersonation attack and stolen smartcard attack. In order to solve these vulnerabilities, Srinivas *et al.* proposed a novel authentication and key agreement scheme for WSN.

In this paper, some weaknesses of Srinivas *et al.*'s scheme are further presented, such as attacks on smart cards, problems in clock synchronization, the lack of resistance to known attacks. In addition, Srinivas *et al.*'s scheme was also vulnerable to node capture attacks and failed to achieve user anonymity. Then an improved user identity authentication protocol IUIAP is proposed based on elliptic curve cryptosystem for multi-gateway WSN. The security of this new protocol is proved using BAN logic [3, 6]. Meanwhile, the performance comparison and efficiency analysis are carried out. The performance analysis proves that the improved protocol IUIAP is suitable for WSN with higher security and little more computation cost.

2 Review of Srinivas *et al.*'s Scheme

In this section, Srinivas *et al.*'s scheme is reviewed briefly. Srinivas *et al.*'s scheme consists of six phases: the system setup phase, the registration phase (sensor node registration phase and user registration phase), the system environment phase, the login, authentication and key agreement phase, the dynamic node addition phase and the password change phase. All the notations used in this paper are listed in Table 1.

Table 1: All the notations used in this paper

Symbols	Quantity
U_i	i^{th} remote user
SN_i	j^{th} sensor node
GWN	gateway node (base station) in WSNs
SC	smartcard of user U_i
ID_i	identity of user U_i
ID_j	identity of sensor node SN_j
r_i, r_j, r_h, r_f	specific magnetization secret nonces used by U_i, SN_j, HGWN and FGWN respectively
X_{GWN}^H	master secret key of HGWN
X_{GWN}^F	master secret key of FGWN
PW_i	password of user U_i
SK	session key shared between U_i and SN_i
$A\|\|B$	concatenation of data A with date B
$A \oplus B$	exclusive-OR of data A with date B
$h(\cdot)$	secure collison-free cryptographic one-way hash function

2.1 System Setup Phase

System setup phase is carried out via the system administrator (SA), which is in an off-line mode. First of all, SA form the identities ID_j for each node. Second, SA chooses a random number that all gateway nodes are know it and computes $P_j = h(ID_j \oplus S_{ran})(i < j < m)$, SA store $\{ID_j, P_j\}$ into the sensor node. Then the sensor node is deployed over a target region.

2.2 Registration Phase

Sensor node registration phase:

(R1). SN_j send $< ID_j, T_r, M_j >$ to the gateway node, which $M_j = h(ID_j||P_j||T_r)$ (T_r is the registration time);

(R2). HGWN computes $P_j = h(ID_j \oplus S_{ran})$, checks it that whether $M_j = h(ID_j||P_j||T_r)$ or not. If confirmed equal, HGWN sends this massage to SN_j, and stores $\{ID_j, T_r\}$.

(R3). SN_j saves T_r.

User registration phase

(R1). U_i chooses ID_i and PW_i. Then U_i generates a random number u. U_i computes $DID_i = h(ID_i||u)$ and $RPW_i = h(PW_i||u||ID_i)$, and gives $< DID_i, RPW_i >$ to HGWN.

(R2). HGWN checks DID_i whether it is registered or not. If it is not registered, HGWN chooses a random number TID_i, computes $K_i = h(DID_i||TID_i||X^H_{GWN})$ and $Y_i = K_i \oplus RPW_i$, where X^H_{GWN} is the secret key of HGWN.

(R3). HGWN stores $\{TID_i, DID_i\}$ in the database, and writes $\{Y_i, TID_i, h(\cdot)\}$ to the smart card SC_i. Then HGWN issues SC_i to U_i.

(R4). U_i inputs the biometric key B_i and computes $C_i = u \oplus H(B_i)$ and $V_i = h(ID_i \oplus PW_i \oplus u)$. U_i stores C_i and V_i into SC_i.

2.3 Login Phase

(L1). U_i inputs ID_i, PW_i and B_i, computes $u = SC_i \oplus H(B_i)$ and compares and calculates whether $h(ID_i \oplus PW_i \oplus u)$ is equal to V_i in the smart card. If equal, calculate $K_i = Y_i \oplus h(PW_i||u||ID_i)$.

(L2). SC_i inquires for sensor identity and HGWN sends the available sensor's identity ID_j to U_i.

(L3). SC_i selects a random number r_i and computes $DID_i = h(ID_i||u)$, $D_1 = h(K_i||DID_i||ID_j)$ and $D_2 = h(DID_i||r_i||TID_i||K_i||T_1||ID_j)$. T_1 is the current timestamp.

(L4). U_i sends the login message $M_1 =< TID_i, ID_j, D_1, D_2, T_1 >$ to HGWN.

2.4 Authentication Phase

Case 1.

(A1). HGWN receives the login message $M_1 =< TID_i, ID_j, D_1, D_2, T_1 >$, and checks the validity of T_2. If verification is through, HGWN used an alias to identify TID_i and extracted DID_i from its database. HGWN computes $K'_i = h(DID_i||TID_i||X^H_{GWN})$ and obtains $r_i = D_1 \oplus h(K_i||DID_i||ID_j)$. Then HGWN verifies whether D'_2 is equal to $h(DID_i||r_i||TID_i||K_i||T_1||ID_j)$ or not.

(A2). HGWN generates a random number r_h, computes $P_j = h(ID_j \oplus S_{ran})$, $D_3 = h(P_j||T_2||T_r||ID_j||TID_j) \oplus r_h$, $D_4 = r_i \oplus h(P_j||TID_i||r_h||T_2)$,$D_5 = DID_i \oplus h(P_j||r_i||r_h||T_2)$, $D_6 = h(ID_{SN_j}||r_i||DID_i||r_h||T_2)$, HGWN sends $M_2 =< TID_i, D_3, D_4, D_5, D_6, T_2 >$ to SN_j.

(A3). SN_j checks the validity of T_3. If the validation is valid, SN_j computes $r'_h = h(P_j||T_2||T_r||ID_j||TID_i) \oplus D_3$, $r'_i = D_4 \oplus h(P_j||TID_i||r_h||T_2)$, $DID'_i = D_5 \oplus h(P_j||r_i||r_h||T_2)$, Then SN_j verifies whether D_6 is equal to $h(ID_j||r_i||DID_i||P_j||r_h||T_2)$ or not. If the verification does not hold, the connection is aborted.

(A4). SN_j generates a random number r_j, computes $D_7 = r_j \oplus h(P_j||r_j||T_3)$, $D_8 = h(P_j||r_j||T_2||r_h||TID_i||r_i||T_3||T_r)$, SN_j sends $M_3 =< D_7, D_8, T_3 >$ to HGWN.

(A5). HGWN checks T_4. If the verification is through, HGWN computes $r'_i = D_7 \oplus h(P_j||r_h||T_3)$. Then HGWN verifies whether D_8 is equal to $h(P_j||r'_j||T_2||r_h||TID_i||r_i||T_3||T_r)$ or not. If the verification does not hold, the connection is aborted. Or HGWN computes $D_9 = r_h \oplus h(K_i||DID_i||r_i)$, $D_{10} = r_i \oplus h(K_i||r_h||DID_i||r_i)$, $D_{11} = h(K_i||DID_i||r_i||T_1||T_h||T_4||r_i)$. Then HGWN sends $M_4 =< D_9, D_{10}, D_{11}, T_4 >$ to U_i.

(A6). U_i checks T_5. If the validation is valid, U_i computes $r_h = D_9 \oplus h(K_i||DID_i||r_i)$, $r_j = D_{10} \oplus h(K_i||r_h||DID_i||r_i)$ and verifies whether D_{11} is equal to $h(K_i||DID_i||r_i||T_1||T_h||T_4||r_i)$ or not. If they are equal, U_i computes the session key $SK = h(DID_i||r_i||r_j||r_h||ID_j)$.

Case 2. On verifying the identity of the sensor node, if this verification failed, HGWN sends a broadcast message $< ID_j, TID_j, ID_h, L_1, T_H >$ to the rest of the gateway nodes.

(A7). FGWN receives the message $< ID_j, TID_j, ID_h, L_1, T_H >$, and verifies the success of T'_H. If this verification succeeds, FGWN verifies whether L_1 is equal to $h(ID_j||TID_i||ID_h||S_{ran}||T_H)$ or not. If it is equally, FGWN calculates $K^F_i = h(TID_i||X^F_{GWN})$,$A_1 = h(TID_i||S_{ran}||ID_f) \oplus K^F_i$, $L_2 = h(TID_i||ID_f||K^F_i||ID_j||T_F)$. FGWN sends the message $M_5 =< A_1, ID_f, L_2, TID_i, T_F >$ to HGWN.

(A8). HGWN receives the message $M_5 =< A_1, ID_f, L_2, TID_i, T_F >$, and checks the validity of T_u. If the verification is passed, calculates $K^F_i = A_1 \oplus h(TID_i||S_{ran}||ID_f)$, and verifies whether

L_2 is equal to $h(TID_i||ID_f||K_i^F||ID_j||T_F)$ or not. If it is equally, HGWN computes $K_i = h(DID_i||TID_i||X_{GWN}^H)$, $A_2 = K_i^F \oplus h(K_i||ID_f)$, $L_3 = h(K_i||K_i^F||ID_j||ID_f||T_u)$. Then sends the message $M_6 =< A_2, L_3, ID_f, T_u >$ to U_i.

(A9). U_i checks the T_2, computes $K_i^F = A_2 \oplus h(K_i||ID_f)$, and verifies whether L_3 is equal to $h(K_i||K_i^F||ID_j||ID_f||T_u)$ or not. If it is equally, U_i computes $A_3 = h(K_i^F||TID_i||ID_j) \oplus r_i$, $L_4 = h(TID_i||r_i||K_i^F||T_2||ID_j)$. Then U_i sends the message $M_7 =< TID_i||ID_j||A_3||L_4||T_2 >$ to FGWN.

(A10). FGWN checks the freshness of T_3. If the verification is though, FGWN computes $K_i^F = h(TID_i||X_{GWN}^F)$ and $r_i = A_3 \oplus h(K_i^F||TID_i||ID_j)$. FGWN verifies whether L_4 is equal to $h(TID_i||r_i||K_i^F||T_2||ID_j)$ or not. If it is equally, FGWN is produced a random number r_f, computes $P_j = h(ID_j \oplus S_{ran})$, $A_4 = h(P_j||T_3||T_r||ID_j||TID_i) \oplus r_f$, $A_5 = r_i \oplus h(P_j||TID_i||r_f||T_3)$ and $L_5 = h(ID_j||r_i||TID_i||T_r||P_j||r_f||T_3)$. Then FGWN sends the message $M_8 =< TID_i, A_4, A_5, L_5, T_3 >$ to SN_j.

(A11). SN_j checks the timestamp T_4. If the verification passed, SN_j computes $r_f = A_4 \oplus h(P_j||T_3||T_r||ID_j||TID_i)$, $r_i = A_5 \oplus h(P_j||TID_i||r_f||T_3)$, and verifies whether L_5 is equal to $h(ID_j||r_i||TID_i||T_r||P_j||r_f||T_3)$ or not. If the verification holds, SN_j is produced a random number r_j and computes $A_6 = r_j \oplus h(P_j||r_f||T_4)$, $L_6 = h(P_j||r_j||T_3||r_f||TID_i||r_i||T_4||T_r)$. Then SN_jsends the message $M_9 =< A_6, L_6, T_4 >$ to FGWN.

(A12). FGWN checks the timestamp T_5, If the verification passed, FGWN computes $r_j = A_6 \oplus h(P_j||r_f||T_4)$, and verifies whether L_6 is equal to $h(P_j||r_j||T_3||r_f||TID_i||r_i||T_4||T_r)$ or not. If the verification holds, FGWN calculates $A_7 = r_f \oplus h(K_i^F||TID_i||r_i)$, $A_8 = r_j \oplus h(K_i^F||r_f||TID_i||r_i)$, and $L_7 = h(K_i^F||TID_i||r_j||T_2||r_f||T_5||r_i)$. Then FGWN sends the message $M_{10} =< A_7, A_8, L_7, T_5 >$ to U_i.

(A13). U_i checks T_6. If the verification is though, U_i extracts $r_f = A_7 \oplus h(K_i^F||TID_i||r_i)$, $r_j = A_8 \oplus h(K_i^F||r_f||TID_i||r_i)$, and verifies whether L_7 is equal to $h(K_i^F||TID_i||r_j||T_2||r_f||T_5||r_i)$ or not. If it is equally, it is confirmed that the sensor node is authentic.

(A14). After successful mutual authentication, the related entities in the system construct a public secret session key $SK = h(TID_i||r_i||r_j||r_f||ID_j)$.

3 Functional and Security Flaws on Srinivas *et al.*'s Scheme

3.1 Smart Card Loss Attack 1

It assumed that the attacker A obtains the biometric B_i and secret key $\{TID_i, C_i, V_i, h(\cdot), Y_i\}$ from the smart card. Therefore, A may achieve identity of U_i and password as follows:

Step 1. A surmises that the identity ID_i^* and the password PW_i^* of U_i.

Step 2. A computes $u = C_i \oplus H(B_i)$ and $V_i^* = h(ID_i^* \oplus PW_i^* \oplus u)$, C_i achieves from the smart card.

Step 3. A tests and verifies whether V_i^* is equal to V_i or not. If they are equally, the guess is correct. Or returns to Step 1.

3.2 Smart Card Loss Attack 2

Step 1. A selects a pair of (ID_i^*, PW_i^*).

Step 2. A calculates $K_i^* = Y_i \oplus h(PW_i^*||u||ID_i^*)$, $u = C_i \oplus H(B_i)$, $DID_i = h(ID_i||u)$, $D_2^* = h(DID_i^*||r_i^*||TID_i||K_i^*||T_1||ID_j)$,

Step 3. A tests and verifies whether D_2^* is equal to D_2 or not. If they are equally, the guess is correct. Or returns to Step 1.

3.3 User Anonymity

User anonymity points that the attack A should be no ability to detect the ID_i of the user by login messages and authentication messages. Srinivas *et al.*'s scheme claimed that the use of pseudoidentity provides user anonymity. But it is not true.When U_i want to access the sensor date, U_i sends the message $M_1 =< TID_i, ID_j, D_1, D_2, T_1 >$ to HGWN. On receiving the login message $M_1 =< TID_i, ID_j, D_1, D_2, T_1 >$, HGWN contracts the login message $M_2 =< TID_i, D_3, D_4, D_5, D_6, T_2 >$ to the sensor nodes. TID_i is a random number and it is a pseudoidentity, and the attacker A can track it to obtain the identity of the user. Therefore, Srinivas *et al.*'s scheme cannot provide un-traceability.

3.4 Node Capture Attack

The biggest threat to the network is the node capture attack. In the node capture attack, the sensor nodes are physically captured, the useful secret information in the nodes is extracted, their programs are modified, and malicious nodes may be used to replace them. By capturing the node, the attacker can obtain the key data in the node storage space, and as the number of captured nodes increases, the attacker will obtain more and more keys until all communication keys are captured. The key system will completely lose security. Srinivas *et al.*'s scheme supposed

that the attacker A can damage all secrets of the captured sensor nodes and asserted that it is impossible for an attacker to obtain three key parameters at a time from the communication information. However, Once the attacker captures the sensor node, he can obtain and calculate the session key. So Srinivas *et al.*'s scheme is not safe.

Step 1. In the m^{th} protocol operation, the messages $\{TID_i, D_3, D_4, D_5, D_6, T_2\}$ sent by HGWN to S_j are intercepted, and the messages $\{D_7, D_8, T_3\}$ sent by S_j to HGWN.

Step 2. The attacker picked up the secret parameter P_j and the registration time T_r.

Step 3. The attacker computes $r_h = h(P_j||T_2||T_r||ID_j||TID_i) \oplus D_3$, $r_i = h(P_j||T_2||r_h||TID_i) \oplus D_4$, $DID_i = h(P_j||r_i||r_h||T_2) \oplus D_5$, $r_i = h(P_j||r_h||T_3)$.

Step 4. The attacker computes the session key $SK = h(DID_i||r_i||r_j||r_h||ID_j)$.

3.5 No Clock Synchronization

The protocol is not affected by clock synchronization and time delay. That is, the server and client do not need to synchronize their clocks with all input devices. In a wireless sensor network, the communication between users and the GWN can be carried out through wired or wireless channels, and the GWN can communicate with sensor nodes through wireless channels. Therefore, the clock synchronization of these three parties is a big challenge for itself. The plan of Srinivas *et al.* uses random numbers and time stamp mechanisms, and their scheme may encounter clock asynchronous problems.

3.6 Resistance to Known Attack (Insider Attack)

In registration phase, suppose A can get the user's password and biometrics and send the information to the HGWN, the scheme will suffer an insider attack.

Step 1. Picks a pair of (ID_i^*, PW_i^*) form the dictionary space.

Step 2. Computes $RPW_i^* = h(PW_i^*||u||ID_i^*)$ and $u = C_i \oplus H(B_i)$.

Step 3. Compares whether $RPW_i^* = RPW_i$ holds.

Step 4. Repeats Steps 1 $\sim$ 3 until find the correct pair of (ID_i^*, PW_i^*).

4 Improved Protocol IUIAP

Improved protocol IUIAP is divided into five phases: system setup phase, registration, login, authentication and password change phase. All the new notations used in the improved protocol IUIAP are listed in the Table 2.

Table 2: All the new notations used in the improved protocol IUIAP

Symbols	Quantity
S_j	j^{th} sensor node
ID_g	identity of foreign gateway node HGWN
ID_f	identity of foreign gateway node FGWN
$f(\cdot)$	Adecoding function
$F(\cdot)$	A fuzzy commitment scheme

4.1 System Setup Phase

First, system administrator (SA) selects a subset of the elliptic curve $E(F_p)$, and chooses a point on the elliptic curve as the base point P. SA selects an identityID_j for each sensor node S_j, and chooses the identity ID_g and the secret key K_{gwn} for HGWN, and computes $K = K_{gwn} \cdot P$, where K is a public key of HGWN. Then SA stores the message $< ID_g, K_{gwn} >$ into HGWN.

4.2 Registration Phase

The registration phase is divided into two phases. One is sensor node registration phase. Another is user registration phase.

1) Sensor node registration phase

In this phase, S_j sends the ID_j as the identity to HGWN by an open channel. On receiving the message, HGWN computes $K_g = h(ID_j||K_{gwn})$. Further, HGWN stores the message $< ID_j, K_g >$.

2) User registration phase

Step 1. User U_i chooses an identity ID_i, password PW_i, and inputs the biometric information B_i and generates a random number a. Then user computes $RPW_i = h(PW_i||a)$. U_i submits the registration quest message $\{ID_i, RPW_i, B_i\}$ to HGWN.

Step 2. When receiving the registration request, HGWN generates a random number b, and computes $F(b, B_i) = (\alpha, \beta)$, $\alpha = h(b)$, and $\beta = B_i \oplus b$. Then HGWN computes $A = h(ID_i||RPW_i||b) \bmod l$, $B = h(ID_i||K_{gwn}||K_g) \oplus h(RPW_i||b)$. where l is an integer between 2^4 and 2^8 as the parameter of fuzzy verifier. HGWN stores the $\{ID_i, b, honey\_list\}$ into its database, and $honey\_list$ is supposed to count the number of failing in user login phase and it is initialized to NULL. Then HGWN stores $< \alpha, \beta, A, B, K, l, f(\cdot), RPW_i >$ into the smart card and sends it to U_i.

4.3 Login Phase

Step 1. U_i inputs the information $\{ID_i, PW_i, B_i'\}$. Then the smart card computes $b' = f(B_i' \oplus \beta) = f(b \oplus (B_i \oplus B_i'))$ and verifies $\alpha \stackrel{?}{=} h(b')$. If they are equal, the session will continue.

Step 2. The smart card computes $A' = h(ID_i||RPW_i||b') \bmod l$ and verifies whether A' is equal to A. If the verification is failed, the session will be over. Or the smart card generates two random numbers c and r_i, computes $K_1 = cP$, $K_2 = cK$, where c multiply the horizontal and vertical coordinates of P and K. $M_1 = h(ID_i||K_{gwn}||K_g) = B \oplus h(RPW_i||b')$, $DID_i = h(K_1||K_2) \oplus ID_i$, $M_2 = M_1 \oplus r_i$, $M_3 = ID_j \oplus h(ID_i||K_2)$, $M_4 = h(M_1||K_1||K_2||M_2||ID_j||r_i)$.

Step 3. U_i sends $< M_2, M_3, M_4, DID_i, K_1 >$ into HGWN.

4.4 Authentication Phase

When HGWN receives login request message, it checks whether ID_j is in the database. If ID_j is in the database, Case 1 will be done.

Case 1.

(A1). HGWN computes $K_2' = K_{gwn}K_1 = cK_{gwn}P$, $ID_i' = h(K_1||K_2') \oplus DID_i$ and checks whether ID_i' in the database. HGWN find *honey_list* by ID_i'. If *honey_list* $\geq$ the preset value, HGWN will rejects the request and sets *honey_list* = *honey_list* + 1. Once passed the inspection, HGWN calculates $M_1' = h(ID_i'||K_{gwn})$, $ID_j' = M_3 \oplus h(ID_i'||K_2')$, $r_i' = M_2 \oplus M_1'$, $M_4' = h(M_1'||K_1||K_2'||M_2||ID_j'||r_i')$ and checks whether M_4' is equal to M_4. The session is rejected by HGWN if they are not equal and updates *honey_list* = *honey_list*+1. Otherwise, HGWN authenticates the user U_i. HGWN generates a random number r_g and computes $K_g' = h(ID_j'||K_{gwn})$, $M_5 = h(K_g'||r_g) \oplus ID_i'$, $M_6 = r_g \oplus K_g'$, $M_7 = h(K_g'||ID_j'||ID_i'||r_g)$. HGWN sends the message $< M_5, M_6, M_7, K_1, K_g' >$ to S_j.

(A2). S_j receives the message and computes $ID_i'' = M_5 \oplus h(K_g'||r_g)$, $r_g = M_6 \oplus K_g'$, $M_7' = h(K_g'||ID_j'||ID_i''||r_g)$. Then S_j checks whether M_7' is equal to M_7. If the verification does not hold, the connection is aborted. Otherwise, S_j generates random numbers r_j and m. Then S_j computes $K_3 = mP$, $K_j = mK_1$, $M_8 = r_j \oplus h(K_g'||K_j)$, $M_9 = h(K_g'||K_j||r_j||ID_i''||ID_j'||K_3)$, $SK_j = h(ID_i''||K_j||K_3||K_1||ID_j')$. Then, S_j sends the message $< M_8, M_9, K_3, K_1 >$ to HGWN.

(A3). HGWN receives the message from S_j and computes $K_j' = mK_1 = mcP$, $r_j = M_8 \oplus h(K_g'||K_j')$, $M_9' = h(K_g'||K_j'||r_j||ID_i''||ID_j'||K_3)$. Then HGWN verifies whether M_9' is equal to M_9. The session is rejected if they are not equal. Otherwise, HGWN calculates $M_{10} = h(K_{gwn}||K_3) \oplus r_g$, $M_{11} = h(ID_i''||K_j'||K_3||K_1||SK_j)$. HGWN sends $< M_{10}, M_{11} >$ to U_i.

(A4). When receiving message from HGWN, U_i calculates $M_{11}' = h(ID_i||K_j'||K_3||K_1||SK_j)$. Then U_i Checks whether M_{11}' is equal to M_{11}. If the verification does not hold, the connection is aborted. Otherwise, it is confirmed that the sensor node is authentic. On the success of mutual authentication, session-key $SK_i = h(ID_i||K_j'||K_3||K_1||ID_j)$ is constricted by involved entities in the system.

Case 2.

(S1). HGWN computes $K_2' = K_{gwn}K_1 = cK_{gwn}P$, $ID_i' = h(K_1||K_2') \oplus DID_i$ and checks whether ID_i' in the database. HGWN find *honey_list* by ID_i'. If *honey_list* $\geq$ The preset value, HGWN will rejects the request and sets *honey_list* = *honey_list* + 1. Once passed the inspection, HGWN calculates $M_1' = h(ID_i'||K_{gwn})$, $ID_j' = M_3 \oplus h(ID_i'||K_2')$, $r_i' = M_2 \oplus M_1'$, $M_4' = h(M_1'||K_1||K_2'||M_2||ID_j'||r_i')$ and checks whether M_4' is equal to M_4. The session is rejected by HGWN if they are not equal and updates *honey_list* = *honey_list* + 1. Then HGWN broadcasts $V_1 =< ID_i', ID_j', ID_f, K_{gwn} >$ to other foreign gateway nodes, where ID_f is the identity of FGWN. If a gateway node FGWN searches ID_j' in its database, it generates a random number x, and computes $L_0 = h(ID_i'||K_{gwn}^f)$, $L_1 = h(ID_f||K_{gwn})$, $L_2 = x \oplus h(L_0||L_1)$, $L_3 = h(ID_i'||ID_j'||L_0||L_1||L_2||x)$, where K_{gwn}^f is the secret key of FGWN. Then FGWN sends $V_2 =< L_3, K_{gwn}, L_1, L_2, L_0 >$ to HGWN.

(S2). HGWN generates a random number r_g*, calculates $x = L_2 \oplus h(L_0||L_1)$, checks whether L_3 is equal to $h(ID_i'||ID_j'||L_0||L_1||L_2||x)$. Then computes $L_4 = ID_i' \oplus h(ID_j'||r_g*)$, $L_5 = r_g* \oplus K_{gwn}$, $L_6 = h(K_{gwn}||r_g*||ID_i'||ID_j'||ID_f)$. HGWN sends the message $V_3 =< L_6, L_5, L_4, L_1, L_0 >$ to U_i.

(S3). U_i computes $ID_i'' = L_4 \oplus h(ID_j'||r_g*)$, $r_g* = L_5 \oplus K_{gwn}$, and verifies whether L_6 is equal to $h(K_{gwn}||r_g*||ID_i''||ID_j'||ID_f)$. If verification holds, U_i selects a random number r_i^*, and computes $L_7 = L_1 \oplus r_i^*$, $L_8 = h(ID_i''||r_i^*||ID_f)$. Then U_i sends the message $V_4 =< L_1, L_7, L_8, L_0 >$ to FGWN.

(S4). FGWN computes $r_i^* = L_1 \oplus L_7$, and verifies whether L_8 is equal to $h(ID_i''||r_i^*||ID_f)$. If the verification succeeds, FGWN generates a random number r_f and computes $L_9 = L_0 \oplus r_f$, $L_{10} = h(r_f||K_{gwn}^f||ID_i'') \oplus r_i^*$, $L_{11} = h(r_i^*||r_f||ID_j'||L_0)$. FGWN sends the message $V_5 =< L_{11}, L_{10}, L_9 >$ to S_j.

(S5). S_j computes $r_f = L_0 \oplus L_9$, $r_i^* = h(r_f||K_{gwn}^f||ID_i'') \oplus L_{10}$, and checks whether L_{11} is equal to $h(r_i^*||r_f||ID_j'||L_0)$. If the verification holds, S_j products r_j^*, computes $SK_j^* = h(r_i^*||r_f||r_j^*||ID_j'||ID_i'')$, $L_{12} = r_j^* \oplus h(r_i^*)$, $L_{13} = h(ID_i''||ID_j'||L_0||SK_j^*)$. Then S_j sends the message $V_6 =< L_{12}, L_{13}, SK_j^*, r_f, r_i^* >$ to FGWN.

(S6). FGWN computes $r_j^* = L_{12} \oplus h(r_i^*)$, $SK_j^* = h(r_i^*||r_f||r_j^*||ID_j'||ID_i'')$, and checks whether L_{13} is equal to $h(ID_i''||ID_j'||L_0||SK_j^*)$. If the verification corrects, FGWN computes $L_{14} = r_f \oplus r_j^*$, $L_{15} = h(L_1||ID_i''||SK_j^*||r_f)$. Then FGWN sends the message $V_7 =< L_{12}, L_{14}, L_{15}, r_i^*, r_j^* >$ to U_i.

(S7). U_i computes $r_f = r_j^* \oplus L_{14}$, $r_j^* = L_{12} \oplus h(r_i^*)$, and checks whether L_{15} is equal to $h(L_1||ID_i''||SK_j^*||r_f)$. If it is true, it is confirmed that the sensor node is authentic. The session key $SK_i^* = h(r_i^*||r_f||r_j^*||ID_j'||ID_i'')$ is constructed by involved entities in the system.

4.5 Password Change Phase

U_i inputs $\{ID_i, PW_i, B_i'\}$ on a special device. Then smart card computes $b' = f(B_i' \oplus \beta) = f(b \oplus (B_i \oplus B_i'))$ and checks whether α is equal to $h(b')$. If they are equal, the session will continue. The smart card computes $A' = h(ID_i||RPW_i||b') \bmod l$ and verifies whether A' is equal to A. If the verification is failed, the session will be over. Otherwise, a new password PW_i^* is allowed to be input. The smart card calculates $RPW_i^* = h(PW_i^*||a)$,$A^* = h(ID_i||RPW_i||b') \bmod l$ and $B^* = B_i' \oplus h(RPW_i \oplus b') \oplus h(RPW_i^* \oplus b')$. Finally, the smart card replaces the stored parameters A and B with A^* and B^*.

5 Analysis of Security Properties

5.1 Authentication Proof Using the BAN Logic

According to the analytic procedures of the BAN logic, the proposed protocol IUIAP should satisfy the following goals:

Goal 1: $HGWN \models HGWN \overset{SK}{\longleftrightarrow} U_i$

Goal 2: $HGWN \models U_i \models HGWN \overset{SK}{\longleftrightarrow} U_i$

Goal 3: $U_i \models U_i \overset{SK}{\longleftrightarrow} HGWN$

Goal 4: $U_i \models HGWN \models U_i \overset{SK}{\longleftrightarrow} HGWN$

Goal 5: $HGWN \models HGWN \overset{SK}{\longleftrightarrow} S_j$

Goal 6: $HGWN \models S_j \models HGWN \overset{SK}{\longleftrightarrow} S_j$

Goal 7: $S_j \models S_j \overset{SK}{\longleftrightarrow} HGWN$

Goal 8: $S_j \models HGWN \models HGWN \overset{SK}{\longleftrightarrow} S_j$

Goal 9: $HGWN \models HGWN \overset{K_{gwn}^f}{\longleftrightarrow} FGWN$

Goal 10: $HGWN \models FGWN \models HGWN \overset{K_{gwn}^f}{\longleftrightarrow} FGWN$

Goal 11: $FGWN \models FGWN \overset{K_{gwn}}{\longleftrightarrow} HGWN$

Goal 12: $FGWN \models HGWN \models FGWN \overset{K_{gwn}}{\longleftrightarrow} HGWN$

Goal 13: $S_j \models S_j \overset{SK}{\longleftrightarrow} FGWN$

Goal 14: $S_j \models FGWN \models S_j \overset{SK}{\longleftrightarrow} FGWN$

Goal 15: $FGWN \models FGWN \overset{SK}{\longleftrightarrow} S_j$

Goal 16: $FGWN \models S_j \models FGWN \overset{SK}{\longleftrightarrow} S_j$

Goal 17: $U_i \models U_i \overset{SK}{\longleftrightarrow} FGWN$

Goal 18: $U_i \models FGWN \models U_i \overset{SK}{\longleftrightarrow} FGWN$

Goal 19: $FGWN \models FGWN \overset{SK}{\longleftrightarrow} U_i$

Goal 20: $FGWN \models U_i \models FGWN \overset{SK}{\longleftrightarrow} U_i$

Idealized form the arrangement of proposed scheme to idealized form is as follows:

Message 1: $U_i \rightarrow HGWN$: $\{M_2, M_3, M_4, DID_i, ID_j, K_1\}$: $\{< r_i >_{h(ID_i||K_{gwn})}, < ID_j >_{h(ID_i||K_2)}, \lhd ID_j||r_i\rangle_{h(K_1||K_2)}, DID_i, ID_j, cP\}$

Message 2: $HGWN \rightarrow S_j : \{M_5, M_6, M_7, K_1, K_g'\}$: $\{< ID_j' >_{h(K_1||K_2)}, < r_g >_{K_g'}, < ID_i'||ID_j' >_{r_g,K_g'}, cP, < ID_j' >_{K_{gwn}}\}$

Message 3: $S_j \rightarrow HGWN : \{M_8, M_9, K_3, K_1\}$: $\{< r_j >_{h(K_g'||K_j)}, < ID_i'||ID_j' >_{K_g',K_j,K_3,r_j}, mP, cP\}$

Message 4: $HGWN \rightarrow U_i : \{M_{10}, M_{11}\}$: $\{< r_g >_{h(K_3||K_{gwn})}, < ID_i' >_{K_j',K_3,K_1,SK_j}\}$

Message 5: $HGWN \rightarrow FGWN : \{ID_i', ID_j', ID_f, K_{gwn}\}$

Message 6: $FGWN \rightarrow HGWN : \{L_3, K_{gwn}, L_1, L_2, L_0\}$: $\{< ID_i'||K_{gwn}^f >_{h(ID_f||K_{gwn})}, K_{gwn}, < K_{gwn} >_{ID_f}, < x >_{h(L_0||L_1)}, < K_{gwn}^f >_{ID_i'}\}$

Message 7: $HGWN \rightarrow U_i : \{L_6, L_5, L_4, L_1, L_0\}$: $\{< ID_i' >_{h(ID_j'||r_g^*)}, < r_g^* >_{K_{gwn}}, < ID_i'||ID_j'||ID_f >_{r_g^*,K_{gwn}}, < K_{gwn} >_{ID_f}, < K_{gwn}^f >_{ID_i'}\}$

Message 8: $U_i \rightarrow FGWN : \{L_1, L_7, L_8, L_0\}$: $\{< K_{gwn} >_{ID_f}, < r_i^* >_{h(ID_f||K_{gwn})}, < ID_i'||ID_f >_{r_i^*}, < K_{gwn}^f >_{ID_i'}\}$

Message 9: $FGWN \rightarrow S_j : \{L_{11}, L_{10}, L_9\}$: $\{ID_i', < r_f >_{h(ID_i'||K_{gwn}^f)}, < r_i^* >_{h(r_f||K_{gwn}^f||ID_i')}, < ID_j' >_{h(ID_i'||K_{gwn}^f), r_i^*, r_f}\}$

Message 10: $S_j \rightarrow FGWN : \{L_{12}, L_{13}, SK_j^*, r_f, r_i^*\}$: $\{< r_j^* >_{h(r_i^*)}, < ID_i'||ID_j' >_{h(ID_j'||K_{gwn}^f), SK_j^*}, SK_j^*, r_f, r_i^*\}$

Message 11: $FGWN \rightarrow U_i : \{L_{12}, L_{14}, L_{15}, r_i^*, r_j^*\}$: $\{< r_j^* >_{h(r_i^*)}, < r_f >_{r_j^*}, < ID_i' >_{SK_j^*, r_f, h(ID_f||K_{gwn})}, r_i^*, r_j^*\}$

The following premised can be given to prove the security of proposed protocol:

Q1: $U_i \models \#(r_i, r_g, r_j, r_f)$

Q2: $HGWN \models \#(r_i, r_g, r_j)$

Q3: $FGWN \models \#(r_i, r_j, r_f)$

Q4: $S_j \models \#(r_i, r_g, r_j, r_f)$

Q5: $U_i \models U_i \overset{K_2}{\longleftrightarrow} HGWN$

Q6: $HGWN \models HGWN \overset{K_g'}{\longleftrightarrow} S_j$

Q7: $S_j \models S_j \overset{K_j}{\longleftrightarrow} HGWN$

Q8: $HGWN \models HGWN \overset{K_3}{\longleftrightarrow} U_i$

Q9: $HGWN \models HGWN \overset{K_2}{\longleftrightarrow} FGWN$

Q10: $FGWN \models FGWN \overset{L_3}{\longleftrightarrow} HGWN$

Q11: $HGWN \models HGWN \overset{r_g^*}{\longleftrightarrow} U_i$

Q12: $U_i \models U_i \overset{r_i^*}{\longleftrightarrow} FGWN$

Q13: $FGWN \models FGWN \overset{r_f}{\longleftrightarrow} S_j$

Q14: $S_j \models FGWN \overset{r_j^*}{\longleftrightarrow} S_j$

Q15: $FGWN \models FGWN \overset{r_f}{\longleftrightarrow} U_i$

P1: $U_i \models HGWN \Rightarrow r_g, r_g^*$

P2: $U_i \models FGWN \Rightarrow r_f$

P3: $S_j \models HGWN \Rightarrow r_g, r_g^*$

P4: $S_j \models FGWN \Rightarrow r_f$

P5: $HGWN \models U_i \Rightarrow r_i, r_i^*$

P6: $HGWN \models S_j \Rightarrow r_j, r_j^*$

P7: $HGWN \models FGWN \Rightarrow K_{gwn}^f$

P8: $FGWN \models S_j \Rightarrow r_j, r_j^*$

P9: $FGWN \models HGWN \Rightarrow K_{gwn}$

P10: $FGWN \models U_i \Rightarrow r_i, r_i^*$

The improved protocol IUIAP can be proved to achieve the secure goals using BAN logic rules.

Message 1:

D1: $HGWN \triangleleft \{< r_i >_{h(ID_i||K_{gwn})}, < ID_j >_{h(ID_i||K_2)}, < ID_j||r_i >_{h(K_1||K_2)}, DID_i, ID_j, cP\}$

D2: According to D1 and Q5, D2 can be got:

$HGWN \mid \equiv U_i \mid \sim \{< r_i >_{h(ID_i||K_{gwn})}, < ID_j >_{h(ID_i||K_2)}, < ID_j||r_i >_{h(K_1||K_2)}, DID_i, ID_j, cP\}$

D3: According to D2 and Q1, freshness conjuncatenation rule and nonce verification rule, D3 can be got:

$HGWN \mid \equiv U_i \mid \equiv \{< r_i >_{h(ID_i||K_{gwn})}, < ID_j >_{h(ID_i||K_2)}, < ID_j||r_i >_{h(K_1||K_2)}, DID_i, ID_j, cP\}$

D4: According to D3, P5, Q8 and jurisdiction rule, D4 can be got:

$HGWN \mid\equiv \{< r_i >_{h(ID_i||K_{gwn})}, < ID_j >_{h(ID_i||K_2)}, < ID_j||r_i >_{h(K_1||K_2)}, DID_i, ID_j, cP\}$

D5: According to D3 and session key rule, D5 can be got:

$HGWN \models HGWN \overset{SK}{\longleftrightarrow} U_i$ (Goal 1)

D6: Using D5 and P5, D6 can be got:

$HGWN \models U_i \models HGWN \overset{SK}{\longleftrightarrow} U_i$ (Goal 2)

Message 2:

D7: $S_j \triangleleft \{< ID_j' >_{h(K_1||K_2)}, < r_g >_{K_g'}, < ID_i'||ID_j' >_{r_g, K_g'}, cP, < ID_j' >_{K_{gwn}}\}$

D8: Using Q6, D7 and message meaning rule, D8 can be got:

$S_j \mid\equiv HGWN \mid \sim \{< ID_j' >_{h(K_1||K_2)}, < r_g >_{K_g'}, < ID_i'||ID_j' >_{r_g, K_g'}, cP, < ID_j' >_{K_{gwn}}\}$

D9: According to Q4, D8, the freshness conjuncatenation rule and nonce verification rule, D9 can be got:

$S_j \mid\equiv HGWN \mid\equiv \{< ID_j' >_{h(K_1||K_2)}, < r_g >_{K_g'}, < ID_i'||ID_j' >_{r_g, K_g'}, cP, < ID_j' >_{K_{gwn}}\}$

D10: From D9, Q7, P3 and the jurisdiction rule, D10 can be obtained:

$S_j \mid \equiv \{< ID_j' >_{h(K_1||K_2)}, < r_g >_{K_g'}, < ID_i'||ID_j' >_{r_g, K_g'}, cP, < ID_j' >_{K_{gwn}}\}$

D11: From D9, Q4 and the session key rule, D11 can be obtained:

$S_j \models S_j \overset{SK}{\longleftrightarrow} HGWN$ (Goal 7)

D12: From D11, Q4 and nonce verification rule, D12 can be obtained:

$S_j \models HGWN \models HGWN \overset{SK}{\longleftrightarrow} S_j$ (Goal 8)

Message 3:

D13: $HGWN \triangleleft \{< r_j >_{h(K'_g||K_j)}, < ID'_i||ID'_j >_{K'_g,K_j,K_3,r_j}, mP, cP\}$

D14: Using Q7, D13 and the message meaning rule, D14 can be got:

$HGWN \mid \equiv S_j \mid \sim \{< r_j >_{h(K'_g||K_j)}, < ID'_i||ID'_j >_{K'_g,K_j,K_3,r_j}, mP, cP\}$

D15: From Q2, D14, the freshness conjuncatenation rule and nonce verification rule, D15 can be got:

$HGWN \mid \equiv S_j \mid \equiv \{< r_j >_{h(K'_g||K_j)}, < ID'_i||ID'_j >_{K'_g,K_j,K_3,r_j}, mP, cP\}$

D16: According to D15, P6 and the jurisdiction rule, D16 can be obtained:

$HGWN \mid \equiv \{< r_j >_{h(K'_g||K_j)}, < ID'_i||ID'_j >_{K'_g,K_j,K_3,r_j}, mP, cP\}$

D17: From Q2, D15 and the session key rule, D17 can be obtained:

$HGWN \models HGWN \overset{SK}{\longleftrightarrow} S_j$ (Goal 5)

D18: From Q2, D17 and the nonce verification rule, D18 can be obtained:

$HGWN \models S_j \models HGWN \overset{SK}{\longleftrightarrow} S_j$ (Goal 6)

Message 4:

D19: $U_i \triangleleft \{< r_g >_{h(K_3||K_{gwn})}, < ID'_i >_{K'_j,K_3,K_1,SK_j}\}$

D20: From D19, Q8 and the message meaning rule, D20 can be obtained:

$U_i \mid \equiv HGWN \mid \sim \{< r_g >_{h(K_3||K_{gwn})}, < ID'_i >_{K'_j,K_3,K_1,SK_j}\}$

D21: From Q1, D20, the freshness conjuncatenation rule and nonce verification rule, D21 can be obtained:

$U_i \mid \equiv HGWN \mid \equiv \{< r_g >_{h(K_3||K_{gwn})}, < ID'_i >_{K'_j,K_3,K_1,SK_j}\}$

D22: According to P1, D21 and the jurisdiction rule, D22 can be obtained:

$U_i \models \{< r_g >_{h(K_3||K_{gwn})}, < ID'_i >_{K'_j,K_3,K_1,SK_j}\}$

D23: According to Q1, D21 and the session key rule, D23 can be obtained:

$U_i \models U_i \overset{SK}{\longleftrightarrow} HGWN$ (Goal 3)

D24: From D23, Q1 and the nonce verification rule, D24 can be obtained:

$U_i \models HGWN \models U_i \overset{SK}{\longleftrightarrow} HGWN$ (Goal 4)

Message 5:

D25: $FGWN \triangleleft \{ID'_i, ID'_j, ID_f, K_{gwn}\}$

D26: From D25, Q9 and the message meaning rule, D26 can be obtained:

$FGWN \models HGWN| \sim \{ID'_i, ID'_j, ID_f, K_{gwn}\}$

D27: From D26 and the nonce verification rule, D27 can be obtained:

$FGWN \models HGWN \models \{ID'_i, ID'_j, ID_f, K_{gwn}\}$

D28: From D27, P9 and the jurisdiction rule, D28 can be obtained:

$FGWN \models \{ID'_i, ID'_j, ID_f, K_{gwn}\}$

D29: According to D27, D28 and the session key rule, D29 can be obtained:

$FGWN \models FGWN \overset{K_{gwn}}{\longleftrightarrow} HGWN$ (Goal 11)

D30: According to D29 and the nonce verification rule, D30 can be obtained:

$FGWN \models HGWN \models FGWN \overset{K_{gwn}}{\longleftrightarrow} HGWN$ (Goal 12)

Message 6:

D31: $HGWN \triangleleft \{< ID'_i||K^f_{gwn} >_{h(ID_f||K_{gwn})}, K_{gwn}, < K_{gwn} >_{ID_f}, < x >_{h(L_0||L_1)}, < K^f_{gwn} >_{ID'_i}\}$

D32: From D31, Q10 and the message meaning rule, D32 can be obtained:

$HGWN \models FGWN| \sim \{< ID'_i||K^f_{gwn} >_{h(ID_f||K_{gwn})}, K_{gwn}, < K_{gwn} >_{ID_f}, < x >_{h(L_0||L_1)}, < K^f_{gwn} >_{ID'_i}\}$

D33: From D32 and the nonce verification rule, D33 can be obtained:

$HGWN \models FGWN \models \{< ID'_i||K^f_{gwn} >_{h(ID_f||K_{gwn})}, K_{gwn}, < K_{gwn} >_{ID_f}, < x >_{h(L_0||L_1)}, < K^f_{gwn} >_{ID'_i}\}$

D34: From D33, P7 and the jurisdiction rule, D34 can be obtained: $HGWN \models \{< ID'_i||K^f_{gwn} >_{h(ID_f||K_{gwn})}, K_{gwn}, < K_{gwn} >_{ID_f}, < x >_{h(L_0||L_1)}, < K^f_{gwn} >_{ID'_i}\}$

D35: According to D34, D33 and the session key rule, D35 can be obtained:

$HGWN \models HGWN \overset{K^f_{gwn}}{\longleftrightarrow} FGWN$ (Goal 9)

D36: According to D35 and the nonce verification rule, D36 can be obtained:

$HGWN \models FGWN \models HGWN \overset{K^f_{gwn}}{\longleftrightarrow} FGWN$ (Goal 10)

Message 7:

D37: $U_i \triangleleft \{< ID'_i >_{h(ID'_j||r^*_g)}, < r^*_g >_{K_{gwn}}, < ID'_i||ID'_j||ID_f >_{r^*_g,K_{gwn}}, < K_{gwn} >_{ID_f}, < K^f_{gwn} >_{ID'_i}\}$

D38: From D37, Q12 and the message meaning rule, D38 can be obtained:

$U_i \mid\equiv HGWN \mid \sim \{< ID_i' >_{h(ID_j'||r_g^*)}, < r_g^* >_{K_{gwn}}, < ID_i'||ID_j'||ID_f >_{r_g^*,K_{gwn}}, < K_{gwn} >_{ID_f}, < K_{gwn}^f >_{ID_i'}\}$

D39: From D38, Q3, the freshness conjuncatenation rule and nonce verification rule, D39 can be obtained:

$U_i \mid\equiv HGWN \mid\equiv \{< ID_i' >_{h(ID_j'||r_g^*)}, < r_g^* >_{K_{gwn}}, < ID_i'||ID_j'||ID_f >_{r_g^*,K_{gwn}}, < K_{gwn} >_{ID_f}, < K_{gwn}^f >_{ID_i'}\}$

D40: According to D39, P5 and the jurisdiction rule, D40 can be obtained:

$U_i \mid \equiv \{< ID_i' >_{h(ID_j'||r_g^*)}, < r_g^* >_{K_{gwn}}, < ID_i'||ID_j'||ID_f >_{r_g^*,K_{gwn}}, < K_{gwn} >_{ID_f}, < K_{gwn}^f >_{ID_i'}\}$

D41: According to Q1, Q3, D40 and the session key rule, D41 can be obtained:

$U_i \models U_i \overset{SK}{\longleftrightarrow} HGWN$ (Goal 3)

D42: According to Q1, D41 and the nonce verification rule, D42 can be obtained:

$U_i \models HGWN \models U_i \overset{SK}{\longleftrightarrow} HGWN$ (Goal 4)

Message 8:

D43: $FGWN \triangleleft \{< K_{gwn} >_{ID_f}, < r_i^* >_{h(ID_f||K_{gwn})}, < ID_i'||ID_f >_{r_i^*}, < K_{gwn}^f >_{ID_i'}\}$

D44: From D43, Q15 and the message meaning rule, D44 can be obtained:

$FGWN \mid \equiv U_i \mid \sim \{< K_{gwn} >_{ID_f}, < r_i^* >_{h(ID_f||K_{gwn})}, < ID_i'||ID_f >_{r_i^*}, < K_{gwn}^f >_{ID_i'}\}$

D45: From D44, Q3, the freshness conjuncatenation rule and nonce verification rule, D45 can be obtained:

$FGWN \models U_i \models \{< K_{gwn} >_{ID_f}, < r_i^* >_{h(ID_f||K_{gwn})}, < ID_i'||ID_f >_{r_i^*}, < K_{gwn}^f >_{ID_i'}\}$

D46: According to D45, P10 and the jurisdiction rule, D46 can be obtained: $FGWN \models \{< K_{gwn} >_{ID_f}, < r_i^* >_{h(ID_f||K_{gwn})}, < ID_i'||ID_f >_{r_i^*}, < K_{gwn}^f >_{ID_i'}\}$

D47: According to Q3, D45, D46 and the session key rule, D47 can be obtained:

$FGWN \models FGWN \overset{SK}{\longleftrightarrow} U_i$ (Goal 19)

D48: According to Q3, D47 and the nonce verification rule, D48 can be obtained:

$FGWN \models U_i \models FGWN \overset{SK}{\longleftrightarrow} U_i$ (Goal 20)

Message 9:

D49: $S_j \triangleleft \{ID_i', < r_f >_{h(ID_i'||K_{gwn}^f)}, < r_i^* >_{h(r_f||K_{gwn}^f||ID_i')}, < ID_j' >_{h(ID_i'||K_{gwn}^f),r_i^*,r_f}\}$

D50: From D49, P10, Q6 and the message meaning rule, D50 can be obtained:

$S_j \mid \equiv FGWN \mid \sim \{ID_i', < r_f >_{h(ID_i'||K_{gwn}^f)}, < r_i^* >_{h(r_f||K_{gwn}^f||ID_i')}, < ID_j' >_{h(ID_i'||K_{gwn}^f),r_i^*,r_f}\}$

D51: From D50, Q4, the the freshness conjuncatenation rule and nonce verification rule, D51 can be obtained:

$S_j \mid \equiv FGWN \mid \equiv \{ID_i', < r_f >_{h(ID_i'||K_{gwn}^f)}, < r_i^* >_{h(r_f||K_{gwn}^f||ID_i')}, < ID_j' >_{h(ID_i'||K_{gwn}^f),r_i^*,r_f}\}$

D52: From D51, P4 and the jurisdiction rule, D52 can be obtained:

$S_j \mid \equiv \{ID_i', < r_f >_{h(ID_i'||K_{gwn}^f)}, < r_i^* >_{h(r_f||K_{gwn}^f||ID_i')}, < ID_j' >_{h(ID_i'||K_{gwn}^f),r_i^*,r_f}\}$

D53: From D52, Q4 and the session key rule, D53 can be obtained:

$S_j \models S_j \overset{SK}{\longleftrightarrow} FGWN$ (Goal 13)

D54: According D53, Q4 and nonce verification rule, D54 can be obtained:

$S_j \models FGWN \models S_j \overset{SK}{\longleftrightarrow} FGWN$ (Goal 14)

Message 10:

D55: $FGWN \triangleleft \{< r_j^* >_{h(r_i^*)}, < ID_i'||ID_j' >_{h(ID_j'||K_{gwn}^f),SK_j^*}, SK_j^*, r_f, r_i^*\}$

D56: From D55, Q14 and the message meaning rule, D56 can be obtained:

$FGWN \mid \equiv S_j \mid \sim \{< r_j^* >_{h(r_i^*)}, < ID_i'||ID_j' >_{h(ID_j'||K_{gwn}^f),SK_j^*}, SK_j^*, r_f, r_i^*\}$

D57: From D56, Q3, the freshness conjuncatenation rule and nonce verification rule, D57 can be obtained:

$FGWN \mid \equiv S_j \mid \equiv \{< r_j^* >_{h(r_i^*)}, < ID_i'||ID_j' >_{h(ID_j'||K_{gwn}^f),SK_j^*}, SK_j^*, r_f, r_i^*\}$

D58: According to D57, P8 and the jurisdiction rule, D58 can be obtained:

$FGWN \mid \equiv \{< r_j^* >_{h(r_i^*)}, < ID_i'||ID_j' >_{h(ID_j'||K_{gwn}^f),SK_j^*}, SK_j^*, r_f, r_i^*\}$

D59: From Q3, D58 and the session key rule, D59 can be obtained:

$FGWN \models FGWN \overset{SK}{\longleftrightarrow} S_j$ (Goal 15)

D60: According to D59, Q3 and the nonce verification rule, D60 can be obtained:

$FGWN \models S_j \models FGWN \overset{SK}{\longleftrightarrow} S_j$ (Goal 16)

Message 11:

D61: $U_i \triangleleft \{< r_j^* >_{h(r_i^*)}, < r_f >_{r_j^*}, < ID_i' >_{SK_j^*,r_f,h(ID_f||K_{gwn})}, r_i^*, r_j^*\}$

D62: From D61, Q1 and the message meaning rule, D62 can be obtained:

$U_i \mid \equiv FGWN \mid \sim \{< r_j^* >_{h(r_i^*)}, < r_f >_{r_j^*}, < ID_i' >_{SK_j^*, r_f, h(ID_f||K_{gwn})}, r_i^*, r_j^*\}$

D63: From D62 and Q1, the freshness conjuncatenation rule and nonce verification rule, D63 can be obtained:

$U_i \mid \equiv FGWN \mid \equiv \{< r_j^* >_{h(r_i^*)}, < r_f >_{r_j^*}, < ID_i' >_{SK_j^*, r_f, h(ID_f||K_{gwn})}, r_i^*, r_j^*\}$

D64: According to Q1, D63 and the jurisdiction rule, D64 can be obtained:

$U_i \mid \equiv \{< r_j^* >_{h(r_i^*)}, < r_f >_{r_j^*}, < ID_i' >_{SK_j^*, r_f, h(ID_f||K_{gwn})}, r_i^*, r_j^*\}$

D65: According to Q1, D64, D63 and the session key rule, D65 can be obtained:

$U_i \models U_i \overset{SK}{\longleftrightarrow} FGWN$ (Goal 17)

D66: According to D65, Q1 and the nonce verification rule, D66 can be obtained:

$U_i \models FGWN \models U_i \overset{SK}{\longleftrightarrow} FGWN$ (Goal 18)

5.2 Further Security Analysis

User anonymity: In wireless sensor networks, an authentication protocol that satisfies user anonymity means that no third party is aware of the identities of the participants in the session, except for the sender of the message and the base station (such as the gateway node mentioned above). In the proposed protocol, the adversary would not be able to identify the sender of the message or whether several different messages came from the same sender. In the improved protocol IUIAP, although adversary intercepts the login request message $< M_2, M_3, M_4, DID_i, K_1 >$ from the legitimate user, K_2' and ID_i' cannot be computed without the awareness of K_{gwn}. Without knowing K_g', the adversary cannot reveal the user's identity from $ID_i'' = M_5 \oplus h(K_g'||r_g)$. Therefore, the user anonymity can be realized based on mathematics difficult problems in elliptic curve cryptosystem.

Forward secrecy: In the improved protocol IUIAP, $K_1 = cP$, $K_3 = mP$, $K_j = mcP$, $SK_i = h(ID_i||K_j'||K_3||K_1||ID_j)$, $SK_j = h(ID_i''||K_j||K_3||K_1||ID_j')$. Although the adversary obtains the secret key K_{gwn}, K_1 and K_3 cannot be computed without knowing the random number m and c. Therefore, SK_i and SK_j in the former sessions cannot be computed in the polynomial time. The improved protocol IUIAP can realize the forward secrecy.

Resist insider attack: Users may register on different system platforms with the same password. An insider may login to another system disguised as the user by obtaining the user's password. Supposed the adversary gets the message $\{ID_i, RPW_i, B_i'\}$ from the database of HGWN, he does not know the PW_i from RPW_i because RPW_i is encrypted by hash functions and a random number. Therefore, the insider attack can be avoided in the improved protocol IUIAP.

Avoid the clock synchronization: The clock synchronization is a big challenge in wireless sensor networks. In the improved protocol IUIAP, the random number method is used to ensure the freshness of exchanged messages. Therefore, the improved protocol IUIAP does not face the clock synchronization problem.

Resist the stolen smart card attack: In the improved protocol IUIAP, user keeps the information $< \alpha, \beta, A, B, K, l, f(\cdot) >$ in the smart card, where $A = h(ID_i||RPW_i||b) \bmod l$, $B = h(ID_i||K_{gwn}||K_g) \oplus h(RPW_i||b)$. Supposed the adversary stoles the smart card and get the information, he cannot guess the ID_i and PW_i without knowing a and b. Without the identity and password, the adversary cannot impersonate as the legitimate user. Therefore, the improved protocol IUIAP can resist stolen smart card attack.

Provide known key security: In the improved protocol IUIAP, S_j and U_i compute the session key respectively. Every secret parameter has been protected by using hash function. The adversary cannot get the secret parameter to compute the new session key even if obtaining the other session keys successfully. Therefore, the improved protocol IUIAP can keep known key secret.

6 Comparisons with Other Related Schemes

In this section, the security performance and efficiency of the proposed protocol IUIAP is evaluated and compared with other protocols. In Table 3, thirteen kinds of security attribute are listed and analyzed among some related protocols. As shown in Table 3, protocols proposed by Aimand Biswas *et al.* and Srinivas *et al.* fail to provide user anonymity, forward secrecy and cannot resist insider attack, the stolen smartcard attack, and remain the problem of clock synchronization. Fortunately, the proposed protocol IUIAP in this paper provides all the security attributes listed in Table 3. Then, the computational overhead between IUIAP and other schemes during the login and authentication phases is analyzed and compared in Table 4. T_h means the operation time of hash function and T_E indicates point multiplication calculation time of elliptic curve. As shown in Table 4, computational overhead is less differentiating in Case 2 among the related protocols. However, calculation amount will not be reduced in Case 1 in protocol IUIAP. But, such increase is

very limited even for communication entities as computer processing performance continues to increase. In brief, protocol IUIAP is more secure than other protocols with only a small amount of calculation increment.

Table 3: Performance comparison among the proposed scheme and other schemes

Security attribute	Amin *et al.* [1]	Srinivas *et al.* [20]	IUIAP
S1	No	No	Yes
S2	No	No	Yes
S3	No	No	Yes
S4	No	No	Yes
S5	No	No	Yes
S6	Yes	Yes	Yes
S7	Yes	Yes	Yes
S8	Yes	Yes	Yes
S9	Yes	Yes	Yes
S10	Yes	Yes	Yes
S11	Yes	Yes	Yes
S12	Yes	Yes	Yes
S13	Yes	Yes	Yes

S1: User anonymity; S2: Forward secrecy; S3: Resist insider attack;S4: The clock synchronization;S5: Resist the stolen smartcard attack;S6: No password verifier table; S8: Password friendly; S9: No password exposure; S10: Sound repairability; S11: Provision of key agreement; S12: Timely typo detection; S13: Mutual authentication.

7 Conclusion

In this paper, Srinivas *et al.*'s protocol has been reviewed and analyzed. Srinivas *et al.*'s protocol fails to resist smart card attacks and other known attacks. Simultaneously, it fails to achieve user anonymity and has some problems in clock synchronization. In addition, Srinivas *et al.*'s scheme is also vulnerable to node capture attacks. Then an improved user identity authentication protocol IUIAP is proposed based on elliptic curve cryptosystem for multi-gateway WSN. The security of the improved protocol is demonstrated by security analysis using BAN logic. By using the elliptic curve cryptosystem (ECC), hash function, dynamic identity mechanism and Multi-factor authentication mechanism, protocol IUIAP can resist the stolen smart card attacks, avoid problems of time synchronization, realize forward secrecy and provide user anonymity. In addition, insider attack also can be resisted and known keys can be kept secret. By comparing with other agreements, protocol IUIAP is more secure than other protocols with only a small amount of calculation increment. Therefore, protocol IUIAP is secure and suitable for multi-gateway WSN in IoT environments.

Acknowledgments

This research was supported by the National Key R&D Program of China on Blue Granary technology innovation (2019YFD0900805).

References

[1] R. Amin, S. K. H. Islam, G. P. Biswas, M. K. Khan, L. Leng, and N. Kumar, "Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks," *Computer Networks*, vol. 101, pp. 42-62, 2016.

[2] S. Anitha, P. Jayanthi, and V. Chandrasekaran, "An intelligent based healthcare security monitoring schemes for detection of node replication attack in wireless sensor networks," *Measurement*, vol. 167, 2021.

[3] S. Challa *et al.*, "An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks," *Computers and Electrical Engineering*, vol. 69, pp. 534-554, 2018.

[4] T. H. Chen and W. K. Shih, "A robust mutual authentication protocol for wireless sensor networks," *ETRI Journal*, vol. 32, no. 5, pp. 704-712, Oct. 2010.

[5] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1086-1090, Mar. 2009.

[6] R. H. Dong, B. B. Ren, Q. Y. Zhang, and H. Yuan, "A lightweight user authentication scheme based on fuzzy extraction technology for wireless sensor networks," *International Journal of Network Security*, vol. 23, no. 1, 2021.

[7] M. S. Farash, M. Turkanovic, S. Kumari, and M. Hoelbi, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment," *Ad Hoc Networks*, vol. 36, pp. 152-176, Jan. 2016.

[8] P. Gope and T. Hwang, "A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7124-7132, Nov. 2016.

[9] D. He, Y. Gao, S. Chan, C. Chen, and J. Bu, "An enhanced two-factor user authentication scheme in wireless sensor networks," *Ad Hoc & Sensor Wireless Networks*, vol. 10, no. 4, pp. 361-371, 2010.

[10] D. He, N. Kumar, and N. J. I. e. Chilamkurti, "A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks," *Information Sciences*, vol. 321, pp. 263-277, 2015.

[11] A. Jabbari and J. B. Mohasefi, "Improvement of a user authentication scheme for wireless sensor networks based on internet of things security," *Wire-*

Table 4: Comparison regarding computation costs

Schemes	Gateway node	Sensor node	User	Total
Amin *et al.* Case 1 [1]	$8T_h$	$5T_h$	$7T_h$	$20T_h$
Amin *et al.* Case 2 [1]	$7T_h$	$5T_h$	$8T_h$	$20T_h$
Srinivas *et al.* Case 1 [20]	$13T_h$	$6T_h$	$10T_h$	$29T_h$
Srinivas *et al.* Case 2 [20]	$16T_h$	$5T_h$	$10T_h$	$35T_h$
IUIAP Case 1	$8T_E + 11T_h$	$2T_E + 5T_h$	$2T_E + 9T_h$	$6T_E + 25T_h$
IUIAP Case 2	$12T_h$	$5T_h$	$6T_h$	$23T_h$

less Personal Communications, vol. 116, no. 3, pp. 2565–2591, 2020.

[12] Q. Jiang, J. Ma, X. Lu, and Y. Tian, "An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 8, no. 6, pp. 1070-1081, Nov. 2015.

[13] Q. Jiang, S. Zeadally, J. Ma, and D. He, "Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks," *IEEE Access*, vol. 5, pp. 3376-3392, 2017.

[14] M. K. Khan and K. Alghathbar, "Cryptanalysis and security improvements of 'two-factor user authentication in wireless sensor networks'," *Sensors*, vol. 10, no. 3, pp. 2450-2459, Mar. 2010.

[15] M. N. Khan, H. U. Rahman, and M. Z. Khan, "An energy efficient adaptive scheduling scheme (EASS) for mesh grid wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 146, pp. 139-157, Dec. 2020.

[16] C. T. Li, C. Y. Weng, C. C. Lee, "An advanced temporal credential-based security scheme with mutual authentication and key agreement for wireless sensor networks," *Sensors*, vol. 13, no. 8, pp. 589–603, 2013.

[17] Z. Liu, F. Yin, J. Ji, and B. Wang, "Revocable and searchable attribute-based encryption scheme with multi-keyword and verifiability for internet of things," *International Journal of Network Security*, vol. 23, no. 2, 2021.

[18] C. Ma, W. Liang, M. Zheng, and H. Sharif, "A connectivity-aware approximation algorithm for relay node placement in wireless sensor networks," *IEEE Sensors Journal*, vol. 16, no. 2, pp. 515-528, 2016.

[19] D. H. Nyang and M. K. Lee, "Improvement of Das's two-factor authentication protocol in wireless sensor networks," *IACR Cryptology ePrint Archive*, 2009.

[20] J. Srinivas, S. Mukhopadhyay, and D. Mishra, "Secure and efficient user authentication scheme for multi-gateway wireless sensor networks," *Ad Hoc Networks*, vol. 54, pp. 147-169, Jan. 2017.

[21] W. L. Tai, Y. F. Chang, and P. L. Hou, "Security analysis of a three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments," *International Journal of Network Security*, vol. 21, no. 6, 2019.

[22] M. Turkanovic, B. Brumen, and M. Hoelbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion," *Ad Hoc Networks*, vol. 20, pp. 96-112, Sep. 2014.

[23] C. Wang, G. Xu, and J. Sun, "An enhanced three-factor user authentication scheme using elliptic curve cryptosystem for wireless sensor networks," *Sensors*, vol. 17, no. 12, 2017.

[24] D. Wang, W. Li, and P. Wang, "Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4064-4075, Sep. 2018.

[25] M. Wazid, A. K. Das, V. Odelu, N. Kumar, M. Conti, and M. Jo, "Design of secure user authenticated key management protocol for generic IoT networks," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 269-282, Feb. 2018.

[26] L. Xu and F. Wu, "A lightweight authentication scheme for multi-gateway wireless sensor networks under IoT conception," *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3977-3993, Apr. 2019.

[27] X. Xu, Z. Gao, and L. Han, "An efficient compromised nodes detection system in wireless sensor networks," *International Journal of Network Security*, vol. 20, no. 5, 2018.

[28] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 316-323, Jan. 2013.

[29] H. L. Yeh, T. H. Chen, P. C. Liu, T. H. Kim, and H. W. Wei, "A secured authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 11, no. 5, pp. 4767-4779, May. 2011.

[30] X. Zhang, B. Wang, and W. Wang, "A new remote authentication scheme for anonymous users using elliptic curves cryptosystem," *International Journal of Network Security*, vol. 20, no. 2, 2018.

Biography

Liling Cao received the B.S. degree in electronic information science and technology from Central South University in 2004, and M.S. degree in physics electronics from Central South University in 2007, and the Ph.D. degree in testing technology and automation from Tongji University in 2017. Now she is a teacher of Department of Engineering Science and Technology Engineering in Shanghai Ocean University. Her main research is Network security, authentication protocol.

Yu Zhang received her bachelor's degree in mechanical engineering from Huaiyin Institute of Technology in 2019. She received her MS degree in mechanical engineering in Shanghai Ocean University in 2021. Her main research is communication security and Internet of things technology.

Mei Liang received her bachelor's degree in electronics and information engineering from ChuZhou University in 2018. Now, she is a student at the school of engineering, Shanghai Ocean University. Her main research is communication security and Internet of things technology.

Shouqi Cao received his bachelor's degree in mechanical manufacturing technology and equipment from Sichuan University in 1996. He received his MS degree in mechanical manufacturing and automation from Sichuan University in 1999. He received his postdoctoral degree in control science and Engineering in Shanghai University in 2009. Now he is a professor and doctoral supervisor of Department of Engineering Science and Technology Engineering in Shanghai Ocean University. His main research interest is marine Internet of things engineering, Fisheries Engineering and automation technology research.

Identification and Detection of Malicious Traffic in Communication Networks with a Deep Learning Algorithm

Fei Yin

(Corresponding author: Fei Yin)

CNNC Key Laboratory on Nuclear Industry Simulation, China Nuclear Power Operation Technology Corporation, LTD.
No. 1021, Minzu Avenue, East Lake High-tech Development Zone, Wuhan, Hubei 430074, China
Email: fe8770@yeah.net

(Received May 1, 2020; Revised and Accepted May 28, 2022; First Online June 23, 2022)

Abstract

The development of the Internet has brought convenience to users in the process of penetrating daily life; however, corresponding requirements are also put forward for network security, especially for detecting malicious traffic. This paper introduced the traffic feature extraction method and the convolutional neural network (CNN) algorithm among Internet malicious traffic detection algorithms. The particle swarm optimization (PSO) algorithm was adopted to adjust the CNN parameters in the training process in order to avoid falling into the locally optimal solution. We compared simulation experiments on the improved CNN algorithm to compare its performance under three activation functions, relu, sigmoid, and tahn. We also compared the performance of the improved CNN algorithm with support vector machine (SVM) and back-propagation neural network (BPNN) algorithms. It was found that the improved CNN algorithm had the best recognition performance when tahn was used as the activation function and also had better recognition performance than SVM and BPNN algorithms; the improved CNN algorithm could maintain a stable malicious traffic filtering effect when facing increasing traffic.

Keywords: Convolutional Neural Network; Deep Learning; Malicious Traffic; Particle Swarm Optimization

1 Introduction

The rapid development of the Internet has not only improved its performance but also gradually facilitated people's daily life [7]. Because of the convenience, people gradually upload important information to the Internet for storage, and therefore the security of the Internet has become increasingly important [1]. Encryption of data transmitted on the Internet can improve the security of user information, but malicious data can also be spread on the Internet with the help of encrypted traffic. Since malicious data are also encrypted, they are difficult to be intercepted directly, threatening the information security of users [5]. Firewalls, as a traditional network security tool, intercept data according to a pre-defined list. However, with the increase of traffic on the Internet, the fixed and rigid interception mode of firewall begins to be difficult to meet the security needs. In order to improve the security of the Internet, more accurate and efficient identification algorithms are needed to detect malicious traffic.

Li *et al.* [6] combined recurrent neural network RNN with Restricted Boltzmann Machines (RBM) for malicious traffic detection and found that the RNN-RBM model had higher detection accuracy, higher recall rate and lower false alarm rate. Liu *et al.* [8] proposed a malicious traffic detection model with a hierarchical attention mechanism. The experimental results showed that the model performed well on different evaluation indicators and also well in the case of a small amount of samples. Gao *et al.* [2] designed a two-level anomaly detection system based on deep neural networks and association analysis and found that the system had high precision and accuracy in identifying malicious traffic. This paper introduced the traffic feature extraction method and convolutional neural network (CNN) algorithm for Internet malicious traffic detection. The particle swarm optimization (PSO) algorithm was used to adjust the CNN parameters in the training process in order to avoid the CNN algorithm from falling into locally optimal solutions. Then, simulation experiments were carried out on the improved CNN algorithm.

2 Malicious Traffic Identification Algorithm

2.1 Traffic Feature Extraction

When the Internet detects whether traffic is malicious, it is impossible for the detection algorithm to quickly and

completely identify the data because of the large volume of traffic and the encryption of the transmitted traffic [3]. Feature extraction is needed before the detection algorithm defects traffic. PCAP is a common format for traffic data [11]. The extracted features are converted into CSV text format. CSV file is the feature information of PCAP file. The specific process is as follows.

1) Messages in the PCAP file are read line by line to determine whether there is an unterminated stream to match the current message; if not, go to Step 2, and if yes, go to Step 3.

2) A new stream is created as the current unterminated stream, and the current message is added to it. The statistical characteristics of this stream in the CSV file are updated. Then, whether the the PCAP file is read is determined. The CSV file is output if it is read; otherwise, it returns to Step 1.

3) The time interval between the current message and the last message of the unterminated stream is calculated. If the time interval is exceeded, it goes to Step 4; if the time interval is not exceeded, it goes to Step 5.

4) The current unterminated stream is ended. The statistical characteristics of this stream in the CSV file are updated, and then it goes to Step 2.

5) Whether the FIN flag of the current message is one is determined. If it is 1, it goes to Step 4; if it is not 1, the current message is added to the unterminated stream. The statistical characteristics of the stream in the CSV file are updated. Whether the PCAP file is read is determined. If it is read, the CSV file is output; if it is not read, it goes to Step 1.

In the process of traffic feature extraction, messages in the PCAP file [12] are divided into a series of streams by the corresponding streaming rules, and then the features in every stream are counted. In this paper, 19 features were extracted from the streams, including average time interval between adjacent messages, the standard deviation of the time interval between adjacent messages, the total number of bytes of forward (backward) message headers, forward (backward) message transmission efficiency, minimum (maximum) message length, average message length, the variance of message length, the standard deviation of message length, number of FINs, number of SYNs, number of RSTs, number of PSHs, number of ACKs, number of URGs, number of CWEs, and number of ECEs. The above 19 features can be directly observed from the stream of messages, without involving the specific content of the messages. These features are not the encrypted object of the messages, so even the features of the encrypted messages can be normally observed, which is conducive to the identification of disguised malicious traffic.

2.2 Identification of Malicious Traffic

This paper chose a CNN algorithm [4] to identify malicious traffic. Compared with a back-propagation neural network (BPNN) algorithm, a traditional deep learning algorithm, the CNN algorithm obtains the local features of samples through convolutional kernels. The plural convolutional features can be combined into overall features. Figure 1 shows the basic structure of the CNN algorithm.

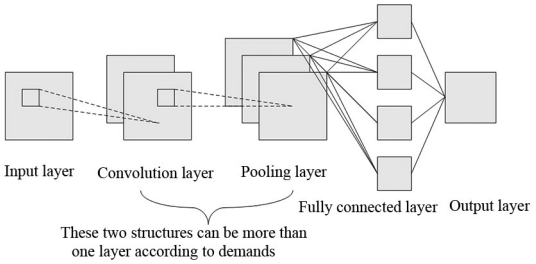


Figure 1: The basic structure of a convolutional neural network

The steps of using the CNN algorithm for traffic identification are as follows.

1) The traffic feature extraction method in 1.1 to convert the traffic PCAP file into a CSV file containing 19 traffic features.

2) Input the CSV file into the input layer of the CNN, and then perform convolutional feature extraction on the sample file using the convolutional kernel in the convolutional layer [8], and the corresponding equation is:

$$x_j^l = f\left(\sum_{j\in M} x_i^{l-1}\cdot W_{ij}^l + b_j^l\right) \tag{1}$$

where x_j^l is the convolutional output feature map, x_i^{l-1} is the feature output of the i-th convolutional kernel in the previous convolutional layer after pooling, W_{ij}^l is the weight parameter between the i-th and j-th convolutional kernels, b_j^l is the bias of j convolutional kernels in l layers, M is the number of convolutional kernels, and $f(\cdot)$ is the activation function.

3) File convolution can be repeated as many times as required, i.e., successive convolution layers can be set up according to the demand, the convolution is followed by pooling of the convolved features in the pooling layer in order to reduce the computation, i.e., the compression pooling operation, including mean-pooling and max-pooling [3]. In mean-pooling and max-pooling, a target frame with a certain size slides on a feature map according to a certain step length, and features in the target frame are compressed. The difference is that mean-pooling averages the features in the target frame as the compression result, while

max-pooling takes the largest feature in the target frame as the compression result.

4) The features are classified in the fully connected layer using the softmax function after multiple convolutions and pooling [13], and the classification results are output in the output layer. If it is in the training phase, the classification results in the output layer are compared with the corresponding labels in training set to calculate the error. Cross-entropy is used as the classification error. Then, the weight parameter and bias term in the convolutional layer are adjusted reversely according to the error. If in the testing phase, the above steps are followed, and the weight and bias parameters in the convolutional layer are set according to the values obtained after training.

2.3 Improvements to the CNN Algorithm

As a forward computing neural network algorithm, the CNN algorithm will adjust the weights according to the error between the calculation result and the actual result during the training.

The typical adjustment method is the gradient descent method, i.e., increasing or decreasing the weight parameter according to a certain step length to reduce the forward calculation error of the CNN algorithm. However, this adjustment method will make the algorithm fall into the locally optimal solution in the training process. In order to get rid of the locally optimal solution in the training process and improve the recognition performance of the CNN algorithm, the PSO algorithm is used to improve the training process of the CNN algorithm.

The improvement principle is to represent the weight and bias parameter in the CNN algorithm with the position of the particles in the PSO population in the multi-dimensional search space, i.e., every particle in the PSO population represents a parameter scheme of the CNN algorithm. The adjustment of the parameters in the CNN algorithm by PSO is to let the plural particles in the population search for the optimal parameter scheme of the CNN algorithm in the search space.

The parameter scheme represented by the particles within the PSO population is substituted during training using the PSO-improved CNN algorithm, and then the forward computation steps are nearly the same as the traditional CNN algorithm. After obtaining the computational error, if the error does not converge to stability or the number of iterations does not reach the limit, the particle velocity and position within the particle population are adjusted using the PSO iteration formula [10]:

$$\begin{cases} v_i(t+1) = \varpi v_i(t) + c_1 r_1 (P_i(t) - x_i(t)) \\ \qquad\qquad + c_2 r_2 (G_g(t) - x_i(t)) \\ x_i(t+1) = x_i(t) + v_i(t+1) \end{cases} \quad (2)$$

where $v_i(t+1)$ and $x_i(t+1)$ are the velocity and position of particle i after one iteration, $v_i(t)$ and $x_i(t)$ are the velocity and position of particle i before the iteration, ϖ is the inertia weight of the particle, c_1 and c_2 are the learning factors, r_1 and r_2 are random numbers between 0 and 1, $P_i(t)$ is the optimal position experienced by particle i, and $G_g(t)$ is the best position experienced by the particle swarm.

The parameter scheme represented by the adjusted particle is substituted into the CNN algorithm again. The CNN algorithm performs forward computation again to obtain the error. Whether to stop training or iterate on the particle swarm again and repeat the forward computation based on the error is determined according to the error.

3 Simulation Experiments

3.1 Experimental Data

The simulation experiments were conducted in a laboratory server using MATLAB software [9] to simulate and analyze the improved CNN algorithm. CIC IoT Dataset 2022 was the dataset used to conduct the simulation experiment, which came from the Canadian Institute for Cybersecurity. Data in the CIC IoT Dataset 2022 were collected from experiments on Internet of Things (IoT) devices. There are six types of data in the dataset, including traffic when power is activated, traffic when the device is idle, traffic when the device is interacting, traffic when the device is running in different scenarios, traffic when the device is active, and traffic when the device is attacked. Then, 1500 normal traffic samples and 1000 malicious traffic samples were selected for subsequent experiments.

3.2 Experimental Setup

The server used for simulation experiments was configured with a Core i7 processor, 16 G memory, and a 64-bit Windows 10 operating system. The PSO-improved CNN algorithm was simulated using MATLAB software. To verify the recognition performance of the improved CNN algorithm, it was compared with SVM and BPNN algorithms.

The relevant parameters of the SVM algorithm are as follows. The sigmoid function was used as the kernel function, and the penalty parameter was set as 1. When the SVM algorithm identifies traffic, it uses the kernel function to map the sample features into a high-dimensional space and then performs calculations to obtain the support vector (hyperplane) to divide the space into two parts. The SVM algorithm is suitable for identifying whether traffic is malicious.

The relevant parameters of the BPNN algorithm are as follows. The number of nodes in the input layer was set as 19, the number of nodes in the output layer was set as 1, the number of nodes in the hidden layer was set as 50, and the tahn function was selected as the activation function in the hidden layer.

The relevant parameters of the PSO-improved CNN algorithm are as follows. The size of both the convolutional kernel and the pooling frame was set as 3, and the convolutional kernel was set as 10 in the convolutional layer. The pooling frame was mean-pooling. The relevant parameters of the PSO algorithm are as follows. The population size was set as 20, both learning factors in the iterative formula were set as 1.3, and the inertia weight was set as 0.7.

The PSO-improved CNN algorithm used the tahn function as the activation function in the convolutional layer, but in addition, relu and sigmoid are also common activation functions; therefore, this paper compared the improved CNN algorithm that used relu, tahn and sigmoid functions.

In addition to the above comparison experiments, this paper also tested the filtering effect of the three detection algorithms on different amounts of malicious traffic. The specific experimental steps are as follows. 500, 1000, 1500, 2000, and 2500 traffic samples were selected from the experimental samples. The proportion of normal traffic and malicious traffic in every sample set was 1.5:1. Three detection algorithms were used to detect and filter sets containing different numbers of traffic samples, and the proportion of malicious traffic in the samples after filtering was recorded.

3.3 Performance Evaluation

The performance evaluation is shown in the following:

$$\begin{cases} P = \frac{TP}{TP+FN} \\ R = \frac{TP}{TP+FP} \\ F = \frac{(\lambda^2+1)\cdot P\cdot R}{\lambda^2\cdot P+R} \end{cases} \tag{3}$$

where P is the precision rate, R is the recall rate, F is the composite value of the precision rate and recall rate, λ reflects the importance of the precision rate and recall rate when evaluating the algorithm (one if both are equally important, greater than one if the recall rate is more important, and less than one if the precision rate is more important), TP indicates the number of malicious samples that are evaluated as malicious, FP indicates the number of malicious samples that are evaluated as normal, FN indicates the number of normal samples that are evaluated as malicious, and TN indicates the number of normal samples that are evaluated as normal.

3.4 Experimental Results

Figure 2 shows the recognition performance of the improved CNN algorithm when different activation functions are used. This paper compared the improved CNN algorithm using three activation functions, relu, sigmoid and tahn. It was seen from Figure 2 that the improved CNN algorithm using relu as the activation function had the worst recognition performance and the improved CNN algorithm using tahn as the activation function had the best recognition performance.

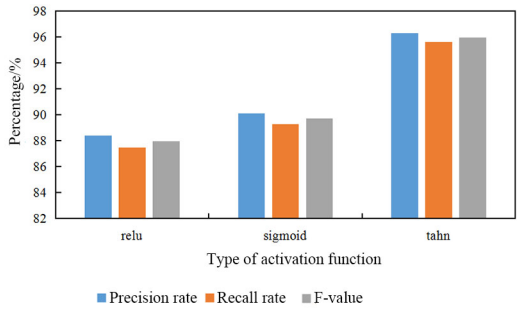


Figure 2: Recognition performance of the improved CNN algorithm using different activation functions

The relu activation function had an output of 0 for negative input and a linear output for positive input; it was faster in training and did not have training gradient saturation when there was positive input, but it was difficult to fit the nonlinear law effectively. The image curve of the sigmoid function was an "S" curve, which was easier to fit the nonlinear law than the relu function, but the gradient of the output value nearly disappeared when the input number was large. Moreover, the output value of the sigmoid function was between 0 and 1, which made it difficult to get a valid cut-off point when judging the binary classification problem. The image curve of the tahn function was also an "S" curve, which had the advantages of taking 0 as the center and the output range between -1 and 1 compared to the sigmoid function, which was more suitable for the binary classification problem.

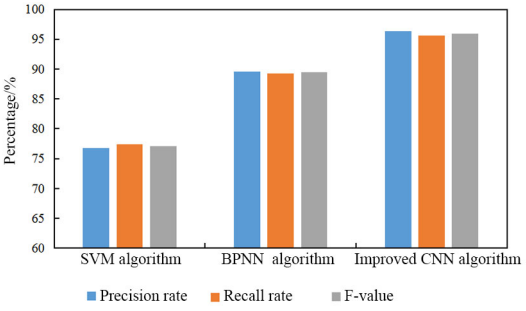


Figure 3: Identification performance of three identification algorithms for malicious traffic

Figure 3 shows the recognition performance of the three recognition algorithms for malicious traffic. The precision rate, recall rate and F-value of the SVM algorithm were 76.8%, 77.4%, and 77.1%, respectively. The precision rate, recall rate and F-value of the BPNN algorithm were 89.6%, 89.3%, and 89.4%, respectively. The precision rate, recall rate and F-value of the improved CNN algorithm were 96.3%, 95.6%, and 95.9%, respectively. It was seen from Figure 3 that the improved CNN algorithm had the highest recognition performance, followed by the BPNN algorithm and the SVM algorithm.

Figure 4 shows the filtering effect of the three identification algorithms after filtering malicious traffic in different

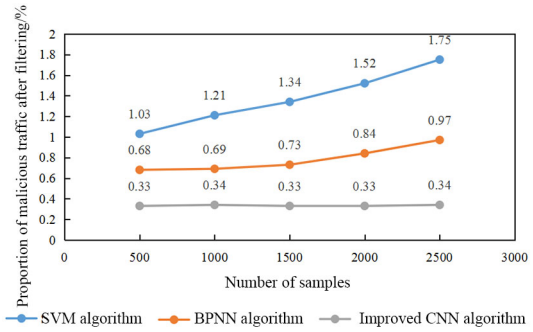


Figure 4: Filtering effect of three identification algorithms on malicious traffic in different numbers of samples

numbers of samples. Before filtering with the identification algorithms, the ratio of normal traffic to malicious traffic in every sample set was 1.5:1, i.e., the proportion of malicious traffic in the sample set before filtering was 40%. It was seen from Figure 4 that the proportion of malicious traffic dropped significantly after filtering the sample set regardless of the identification algorithm; the highest proportion of malicious traffic after filtering was 1.75% under the three algorithms and the five sample quantities. As the number of detected samples increased, the proportion of malicious traffic increased gradually after being filtered by SVM and BPNN algorithms, while the proportion of malicious traffic filtered by the improved CNN algorithm remained stable. The comparison of the filtering effect between different algorithms under the same number of detected samples showed that regardless of the number of samples, the proportion of malicious traffic after filtering by the SVM algorithm was the largest, followed by the BPNN algorithm and the improved CNN algorithm.

The reasons for the above results are as follows. When the SVM algorithm identified traffic, it first mapped the features of the traffic to a high-dimensional space and then used the support vector (hyperplane) obtained from the training to divide the feature space into two parts to separate normal traffic and malicious traffic. The SVM algorithm had a simple principle, but the hyperplane obtained the linear division rule by fitting after mapping the nonlinear features to linear features through the kernel function. Some effective information is inevitable to be missed in the mapping process. The activation function used in the BPNN algorithm could effectively fit the nonlinear law, so it performed better in recognition than the SVM algorithm; however, the activation function in the BPNN algorithm had small variations in the gradient of the output values when facing inputs with relatively large span, which was not conducive to the reverse adjustment during training and was easy to fall into the locally optimal solution. The improved CNN algorithm not only had the advantage of the CNN algorithm, i.e., taking into account both local and overall features, but also used PSO to adjust the parameters in the training process, which avoided the difficult reverse adjustment of parameters caused by the decreased gradient of output values in the later stage of training.

4 Conclusion

This paper introduced the traffic feature extraction method and the CNN algorithm among Internet malicious traffic detection algorithms. Moreover, in order to avoid the CNN algorithm from falling into locally optimal solutions in the training process, the parameters of the CNN algorithm were adjusted using the PSO algorithm. Then, simulation experiments were conducted on the improved CNN algorithm, the performance of the CNN algorithm using relu, sigmoid, and tahn as the activation function was compared, and the performance of SVM, BPNN, and improved CNN algorithms were also compared. The results are as follows. The improved CNN algorithm using tahn as the activation function had the best recognition performance, followed by the algorithm using sigmoid as the activation function and the algorithm using relu as the activation function. In the face of malicious traffic, the improved CNN algorithm had the best recognition performance, the BPNN algorithm had the second-best recognition performance, and the SVM algorithm had the worst recognition performance. When filtering different volumes of traffic, the proportion of malicious traffic after filtering by the improved CNN algorithm was the least, and moreover, the proportion remained stable as the number of samples increased; the proportions of malicious traffic after filtering by SVM and BP algorithms were relatively larger and increased as the number of samples increased.

References

[1] S. H. Almotiri, "Integrated fuzzy based computational mechanism for the selection of effective malicious traffic detection approach," *IEEE Access*, vol. PP, no. 99, pp. 1-1, 2021.

[2] M. Gao, L. Ma, H. Liu, Z. Zhang, Z. Ning, J. Xu, "Malicious network traffic detection based on deep neural networks and association analysis," *Sensors (Basel, Switzerland)*, vol. 20, no. 5, pp. 1-14, 2020.

[3] M. He, X. Wang, J. Zhou, Y. Xi, L. Jin, X. Wang, "Deep-feature-based autoencoder network for few-shot malicious traffic detection," *Security and Communication Networks*, vol. 2021, no. 6, pp. 1-13, 2021.

[4] P. He, G. Gan, "Android malicious app detection based on CNN deep learning algorithm," *IOP Conference Series Earth and Environmental Science*, vol. 428, pp. 012061, 2020.

[5] F. Hussain, S. G. Abbas, G. A. Shah, I. M. Pires, U. U. Fayyaz, F. Shahzad, N. Garcia, E. Zdravevski, "A framework for malicious traffic detection in IoT healthcare environment," *Sensors*, vol. 21, no. 9, 2021.

[6] C. Li, J. Wang, X. Ye, "Using a recurrent neural network and restricted Boltzmann Machines for ma-

licious traffic detection," *NeuroQuantology*, vol. 16, no. 5, pp. 823-831, 2018.

[7] Z. Ling, J. Luo, K. Wu, W. Yu, X. Fu, "TorWard: Discovery, blocking, and traceback of malicious traffic over Tor," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2515-2530, 2017.

[8] X. Liu, J. Liu, "Malicious traffic detection combined deep neural network with hierarchical attention mechanism," *Scientific Reports*, vol. 11, no. 1, 2021.

[9] T. Shibahara, K. Yamanishi, Y. Takata, D. Chiba, T. Hokaguchi, M. Akiyama, T. Yagi, Y. Ohsita, M. Murata, "Event de-noising convolutional neural network for detecting malicious URL Sequences from proxy logs," *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, vol. E101.A, no. 12, pp. 2149-2161, 2018.

[10] H. M. Song, J. Woo, H. K. Kim, "In-vehicle network intrusion detection using deep convolutional neural network," *Vehicular Communications*, vol. 21, no. Jan., pp. 100198.1-100198.13, 2020.

[11] Z. Wang, K. W. Fok, V. Thing, "Machine learning for encrypted malicious traffic detection: Approaches, datasets and comparative study," *Computers & Security*, vol. 113, pp. 102542-, 2022.

[12] H. Yang, Q. He, Z. Liu, Q. Zhang, "Malicious encryption traffic detection based on NLP," *Security and Communication Networks*, vol. 2021, no. 1, pp. 1-10, 2021.

[13] L. Zhang, S. Wang, S. Xiang, J. Zhang, Y. Liang, "Research on the detection method of information system access abnormal behavior," *Journal of Physics: Conference Series*, vol. 1883, no. 1, pp. 012104 (6pp), 2021.

Biography

Fei Yin, born in 1985, has received the master's degree from Jiangxi University of Science and Technology in July 2013. He was major in computer applied technology. He is working in China Nuclear Power Operation Technology Corporation, LTD., as an engineer. He is interested in network security, data security, and industrial Internet security.

Fusion Dilated CNN for Encrypted Web Traffic Classification

Benjamin Appiah[1], Anthony Kingsley Sackey[1], Owusu-Agyemang Kwabena[2], Ansuura JohnBosco Aristotle Kanpogninge[3], and Peter Antwi Buah[4]

(Corresponding author: Benjamin Appiah)

Department of Computer Science, Ho Technical University[1]

Ho, Volta Region, Ghana

Email: bappiah@htu.edu.gh

School of Information and Software Engineering, University of Electronic Science and Technology of China[2]

C. K. Tedam University of Technology, Applied Sciences, Ghana[3]

Department of Civil Engineering, Southwest Jiaotong University, China[4]

(Received Sept. 17, 2020; Revised and Accepted May 19, 2022; First Online June 23, 2022)

Abstract

A Growing number of conventional Convolutional neural network (CNN) models have been employed for encrypted web traffic characterization. However, the application of CNN models is confronted with two significant challenges; a) they possess short reflective fields that don't gather much-encrypted traffic information for effective and accurate predictions. b) these models are not adaptive to the diverse nature of traffic flow because of their single-head architecture. This paper alleviates these problems using the fusion of dilated Convolutional neural networks dubbed FDCNN. FDCNN architecture supports exponentially large receptive fields and captures local dependencies in encrypted traffic data. The experimental results on public datasets, ISCX VPNnon-VPN Traffic datasets, indicate that FDCNN architecture is practical and achieves higher accuracy.

Keywords: Dilated Convolutions; Encrypted Web Traffic; Traffic Characterization

1 Introduction

Encrypted traffic has risen due to the expanding request for online user privacy in today's internet. The downside to achieving this privacy and trust online has lead to difficulty of network administrators to inspect and characterize these encrypted web traffics.Therefore, an accurate characterization of encrypted traffics has become a challenge in modern networks [2].

Recently, a lot of solutions have been proposed to address this challenge, notable of these solutions are those proposed based on Deep Learning (DL), such as [13, 24–26]. These proposed DL classifiers learn on discriminating features embedded in web traffic to make a final prediction, these discriminating features arise from the different pseudo-random generator algorithms employed to encrypt these traffics. However, these DL classifiers face two fundamental problems:

1) they are based on conventional convolutional neural network (CNN) operations. CNN operations have proven to be effective in training and predicting sequential model tasks, however, CNN possesses short receptive fields to gather much information about the training data for effective and accurate predictions [28]. Also, CNN is only able to reference previous information or history with size linear in-depth of the network [3]. These shortfalls in CNN makes it a disadvantage, when applied on sequential task, especially web traffic flows characterization task that requires longer history for effective characterization.

2) these models adopt single-head architecture to process traffic flow as shown in Fig. 1(b), thus, a single feature map containing the main features of all the traffic data is obtained as a result. Even though a different set of convolution filters are used for each layer and therefore, the extraction of features is unrestrained for each layer, all of them are blend to give the final result. In this way, the specific features of each traffic flow data might be lost by mixing them, making the above models not adaptive to new traffic flow configurations [29].

Given the above-stated problems, and motivated by [3, 28, 29], we propose an encrypted web traffic characterization architecture integrating fusion of one dimensional independent dilated Convolutional networks with an Attention mechanism to resolves the two problems of the existing DL-based models. This paper makes the following contributions:

- We introduce an encrypted web traffic characterization architecture named FDCNN, to successfully resolves the two problems of existing DL approaches. This architecture can also be applied to web traffic without the need for feature engineering.
- FDCNN comprises a fusion of one dimensional independent dilated convolutions with an Attention mechanism to support exponentially large receptive field and capture local dependencies in encrypted traffic data.
- We investigate the performance of FDCNN architecture to that of single-head CNN architecture on different encrypted traffic data situations.

Now, we will describe the layout for the remaining of this paper. In the next section, we will briefly review related work. Our proposed approach is described in Section 3. In Section 4, we investigate the performance of FDCNN to that of the single-head CNNs architecture on ISCX VPN-non-VPN datasets [5] and describe the findings. Specifically, the findings demonstrate the superior performance of FDCNN over single-head CNN architecture. Finally, we conclude the paper in the last Section 5.

2 Related Work

With the widespread use of encrypted data transport, web traffic encryption is becoming a standard nowadays. This presents a challenge for traffic measurement, especially for characterization and anomaly detection methods, which are dependent on the type of web traffic. There has been much research in this area which has led to many different proposed techniques [4, 16, 18].

Traditionally, classifying web traffic would be to inspect the traffic payloads. Classifiers use signatures for each protocol which are predefined expressions or patterns which are used to distinguish different traffic protocols [8, 20]. The authors in [20] proposed HTTP Secure (HTTPS) traffic only characterization to solve user privacy issue that can inspect encrypted payload without decryption.

Other forms of characterizing encrypted web traffic data are using well-known port numbers of network protocols such as TCP/UDP [4]. Port-based classifiers distinguish particular applications by comparing application ports numbers with IANA TCP/UDP port numbers. Port-based classifiers are often used in firewalls and access control lists (ACL) [16]. However, since most attackers use the non-standard port to bypass renewals or circumvent operating systems restrictions when conducting attacks, this makes these detection techniques inaccurate for detecting and characterizing encrypted traffic [25]. For these reasons, to characterize modern web traffic more complex traffic classification methods are needed.

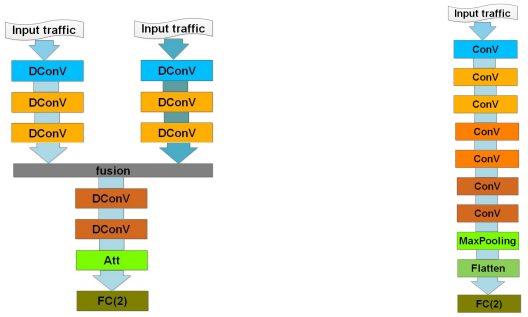
Statistical and machine learning have been looked at to address the encrypted traffic characterization problem. Statistical methods identify traffic using some statistically unique features. Korczynski and Duda [11] proposed Statistical Protocol Identification to classify Skype traffic. In their work, nine traffic flow features were selected. The model was tested on an artificially created dataset including other traffic with HTTP, SSL, SFTP, SCP, BitTorrent, VoIP, SSH. Chanchal Suman *et al.* [22] proposed an unsupervised feature selection technique for analyzing the web data. The proposed model was evaluated on the KDD dataset. However, statistical learning approaches require extensive feature engineering in order to classify web traffic.

Machine learning approaches have been proposed to characterize encrypted web traffic. Mauro *et al.* [4] applied random forest on protocol encapsulated traffic classification. The proposed model uses flow features. Aghaeiet *et al.* [1] proposed C4.5 decision tree classifier on proxy traffic. Artificial neural network (ANN) approaches have also been proposed for encrypted web traffic identification [7, 23]. From the experimentation in [23], the ANN approach outperforms C4.5 and Naive random forest methods. Gil *et al.* [5], proposed k-nearest neighbor (k-NN) and C4.5 decision tree encrypted traffic classifier using time-related features such as flow bytes per second, backward, and forward inter-arrival time, traffic flow duration, etc. to characterize the network traffic. Their work was tested on the ISCX VPNnonVPN dataset.

To the best of our knowledge, prior to our work, Deep learning ideas for encrypted web traffic characterization has been reported by [13, 24–26]. The Authors in [24, 26] proposed stack autoencoders (SAE) and one-dimensional convolution neural networks encrypted traffic classifiers for a large family of protocols like HTTPS, HTTP, SMTP. The work in [24, 26] was evaluated on the ISCX VPN-non-VPN and UNB-CIC Tor Traffic datasets [5, 12] respectively. The Authors in [13, 25] proposed encrypted traffic classification method with one-dimensional convolution neural networks. This method integrates feature extraction, feature selection and classifier into a unified end-to-end framework and were also evaluated on same ISCX VPN-non-VPN dataset. However, efficient deep learning architecture for encrypted web traffic characterization is still a challenge to be addressed. Therefore, the motivation of this work is to investigate new supervised architecture for encrypted web traffic characterization based on fusion dilated convolututional neural networks with an Attention mechanism that can capture, extract meaningful features and characterized encrypted web traffic data.

3 FDCNN Architecture

In this section, a brief overview of our proposed architecture implemented in this article is provide and then we elaborate on the fusion dilated Convolution (FDCNN) and how it is used to address the two problems discussed in Section 1. Figure 1: (a) FDCNN architecture with two



(a) (b)

Figure 1: (a) FDCNN architecture with two independent dilated Convolutional heads. Each dilated convolution (DConv) applies a Batch Normalization (BN) layer followed by ReLU activation layer. The output of the dilated Convolutional heads are fused together and then passed onto lower DConv layers before fed into the Attention layer (Att). (b) Single-head CNNs architecture with stacks of Convolutional networks (ConV) followed by Maxpooling layer (MaxPooling) and Flatten layer (Flatten). Both architectures ends with a 2-way fully connected layers (FC) followed by a Softmax layer

independent dilated Convolutional heads. Each dilated convolution (DConv) applies a Batch Normalization (BN) layer followed by ReLU activation layer. The output of the dilated Convolutional heads are fused together and then passed onto lower DConv layers before fed into the Attention layer (Att). (b) Single-head CNNs architecture with stacks of Convolutional networks (ConV) followed by Maxpooling layer (MaxPooling) and Flatten layer (Flatten). Both architectures ends with a 2-way fully connected layers (FC) followed by a Softmax layer.

3.1 FDCNN Overview

Figure 1(a), show the general overview of our proposed fusion dilated Convolutional neural network with Attention architecture (FDCNN). The FDCNN architecture has two independent dilated Convolutional heads fused together. The outputs of the dilated Convolutional heads are fused together before reaching the lower layer dilated convolutions followed by an Attention mechanism. The FDCNN architecture ends with a 2-way fully connected layer followed by a softmax for classification. A softmax is applied to the resulting volume to convert prediction scores to proper probability distributions.

3.2 Dilated Convolutional Neural Network

Convolutional neural networks have achieved great success in a wide range of problems in the last few years with one of its key notions is the receptive field for effective prediction. A unit in the Convolutional networks only depends on a region of the input sequence. Hence, anywhere in an input sequence outside the receptive field of a unit does not affect the value of that unit. For this reason, it is essential to cautiously control the receptive field, to ensure that it covers the entire relevant sequence regions. For CNN to cover a longer sequence, requires a larger receptive field, which comes at a cost of difficulty in training the model because more layers must be employed to increase the receptive field size [3, 28].

To address the issue of the short reflective field, in FDCNN architecture, we replace CNNs with dilated convolutions [28]. Dilated convolutions as an alternative to the CNN operations offer a robust means to increase the receptive field without increasing the number of parameters or the amount of computation (e.g., adding more Convolutional layers can make larger receptive fields but introduce more operations). In dilated convolution operations, a small-size kernel with $k \times k$ filter is enlarged to $k + (k \times 1)(d \times 1)$ with increasing dilatation factor d ensures that the receptive field of each units are exponentially expanded. Given an encrypted web traffic sequence vector as inputs $x = \{x = x_1, ..., x_T\}, x \in R^T$, Convolutional filter as $f : \{0, ..., k-1\}$. The dilated convolution operation F on elements s of a sequence is defined as:

$$DConV(x) = F(s) = (x_{*d}f)(l) = \sum_{l=0}^{k-1} f(l) \cdot x_{s-d \cdot l} \quad (1)$$

Where d, k and l represents dilation factor, filter size and Convolutional layer respectively and $s - d.l$ accounts for the direction of the past. Since web traffic data can be expressed as a one-dimensional vector, therefore, one-dimensional dilated convolutions are adopted, as a result, the kernel can move over traffic data in a single dimension.

3.3 Fusion Network

In FDCNN architecture as shown in Figure 1(a), we adopt independent dilated convolutions in contrast to the singlehead employed in the literature in Figure 1(b). The independent dilated convolutions allow each model to allocate different upsampling kernels to different components of the traffic data.

For each head H with L layers of dilated convolutions (we consider each independent dilated convolutions as heads), we expand the receptive field by choosing different k's and exponentially increasing dilation by $d = O(2^l)$ at level l in each H. We use different filter sizes and dilation values in each head to ensure that different dynamics in the encrypted traffic data are captured and represent, hence, an independent feature map for each traffic data is obtained.

The final output of each head is interpreted as a fusion of heads, as defined in Equation (2), which are then interpreted by middle layer dilated Convolutional operations.

$$RconnH_{final} = [H_{DConV(x)1} + H_{DConV(x)2}] \quad (2)$$

3.4 Attention Mechanism

Attention mechanisms [14] is proposed to allow modeling of dependencies without regard to their distance in the input or output sequences of the final dilated Convolutional layers. In this work, we replace the max-pooling and flatten layer employed in the single-head Convolutional architectures with dot product attention. Given the intermediate inputs $Y = \{DConV_{RconnHfinal1}, ..., DConV_{RconHfinaln}\}$, dot product attention returns a weighted output which summarizes based on how Y is related to the context vector c and sum to 1, so doing determines the relative contributions of each of the final output. This yields

$$u_i = tanh(W_c \cdot Y_i + b_c) \quad (3)$$

$$\alpha_l = \frac{exp(u_i^T u_c)}{\sum_i exp(u_i^T u_c)} \quad (4)$$

$$z = \sum_i \alpha_i Y_i \quad (5)$$

Where n is the sequence length and W is the weighted matrix. Vector α denotes the attention weights to matrix Y. Matrix z is the representation of the sequence formed by a weighted sum of the output vectors.

3.5 Characterization

We employ a softmax classifier as our final layer. The softmax classifier yields our actual probability scores for each class label Y. For each set of classes Y for an input pattern n, the classifier takes z input with weight W and present

$$\hat{p}(y|n) = softmax(W_{nz} + b_n) \quad (6)$$

$$\hat{y} = argmax_y \hat{p}(y|n) \quad (7)$$

The loss function evaluates how well our proposed model models our datasets. The current error is propagated backward to a previous layer, where it is used to modify weights and bias in such a way that the error is minimized. The Cross-entropy loss function is adopted in our work and it is computed as the differences between the actual output and predicted output.

$$loss = argminJ(y, \hat{y}) \quad (8)$$

$$J(W) = -\sum_i^n y_i log(\hat{y}_i) \quad (9)$$

The function $J(\cdot)$ represent the loss function, y is the output and $\hat{y}$ is the predicted output.

4 Experiments

We compare the performance of FDCNN as describe in Section 3 to single-head Convolutional architecture on encrypted web traffic flow data. Throughout this experimentation, we will abbreviate the single-head Convolutional architecture as SH. We also compare the performance of FDCNN with these state-of-the-art methods that have demonstrated their applicability to the task of encrypted traffic classification such as DeepPacket [13] and DataNet [24].

We implemented FDCNN and SH with Tensorflow and 80% of the performance of the PC, having a 3.60 GHz of i7-477OS CPU, 8.00GB of RAM, GTX 1060 of GPU.

In FDCNN and SH architecture, each convolution applies a Batch Normalization (BN) layer [17] followed by rectified linear unit (ReLU) [15]. Dropout [5] is at 0.1. We use an exponential dilation $d = 2l$ for layer l in each head of the FDCNN network and Adam optimizer [9] with the Mini-Batch of 32 and learning rate set to be in the range of $\{0.01, 0.001, 0.005, 0.0001, 0.0005\}$. The number of filters in the range $\{120, 100, 80, 60, 40, 30, 20\}$ in the Convolutional layers, and kernel size set to be in the range of $\{3, 5\}$. Zero paddings are employed in both architectures to keep before and after layers at the same length. The max-pooling operation with pooling and stride set to 3×3 and 1 respectively in the SH network, in the full connection layer, FDCNN and SH network has neural number set to 400 and 500.

Three evaluation metrics were utilized for performance analysis of our model experiments. These are Precision (P), Recall (R), f-measure (F1), and Confusion matrix. The Precision, Recall and f-measure are described mathematically as follows

$$P = \frac{TP}{TP + FP} \quad (10)$$

$$R = \frac{TP}{TP + FN} \quad (11)$$

$$F1 = \frac{2.P.R}{TP + FP} \quad (12)$$

Where TP represents true positive, FP represent false positive and FN stands for false negative.

Table 1: ISCX VPN-nonVPN Traffic characterization distribution

Traffic characterization	Unbalanced flow Size	Balanced flow Size
CHAT	7848	30000
FTP	12654	30000
MAIL	5172	30000
STREAMING	3683	30000
VOIP	18546	30000
P2P	11415	30000
BROWSING	30000	30000
VPN-CHAT	2839	30000
VPN-P2P	3415	30000
VPN-FT	4704	30000
VPN-MAIL	2444	30000
VPN-STREAMING	1115	30000
VPN-VOIP	5576	30000
VPN-BROWSING	10000	30000
Total traffic	119,412	420,000

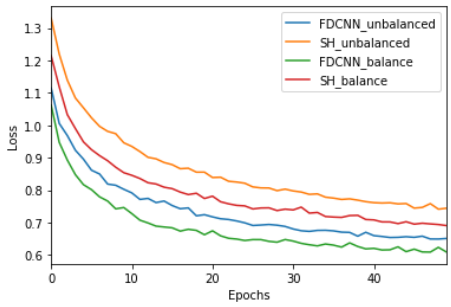


Figure 2: Results on the ISCX VPN-nonVPN Traffic datasets. Epoch = 50. FDCNN outperform SH model

4.1 Datasets and Preprocessing

We evaluate FDCNN and SH on tasks that have been commonly used to benchmark the performance of different SH approaches for encrypted web traffic characterization [13, 24, 25].

We first create two sets of the dataset from each benchmark datasets, i.e. unbalanced and balanced dataset. A comparison of FDCNN to the SH approach is done on the unbalanced and balanced datasets. We normalized benchmark datasets with the Power-Transformer tool [27] and randomly split each dataset into three separate sets: 60% of samples are used for training and adjusting weights and biases, 20% for validation and 20% of data points are used for testing the models.

4.1.1 ISCX VPN-nonVPN Dataset

ISCX VPN-nonVPN dataset [5] represent real-world the traffic characterization summing up to a 14 different classes dataset. This dataset has been used in prior works such as [13,24,25]. Table 1 shows the traffic characterization distribution of ISCX VPN-nonVPN dataset consisting of 7 class regular encrypted traffic and 7 class protocol encapsulated traffics.

Table 2: Weighted average performance of FDCNN and SH. Epochs = 50. The best values are highlighted in bold.

Architecture	P	R	F1
SH-unbalanced	97.31	97.25	97.07
SH-balanced	98.28	97.96	98.18
FDCNN-unbalanced	**98.91**	**98.68**	**98.72**
FDCNN-balanced	**99.31**	**98.65**	**99.32**

4.2 Results and Discussion

Log loss measures the uncertainty of the probabilities of our model by comparing them to the true labels that's measurement of accuracy that incorporates the idea of probabilistic confidence. Log Loss nearer to 0 indicates higher accuracy, whereas if the Log Loss is away from 0 then it indicates lower accuracy. From Figure 2, we can see that perhaps FDCNN may result in better performance or at least lower log loss. FDCNN's log loss is different, going down sooner to a lower value and generally staying lower than the SH on both balanced and unbalanced datasets.

Table 3: FDCNN and SH time occupancy Comparison on the ISCX VPN-nonVPN dataset

Architecture	P
SH-unbalanced	21
SH-balanced	98
FDCNN-unbalanced	73
FDCNN-balanced	259

Table 4: Results using varying kernel size in FDCNN architecture. Epochs=50

k	Average testing accuracy $\pm$ standard deviation (%)
20×1	99.35 ± 0.06
30×1	98.42 ± 0.19
40×1	98.92 ± 0.14
60×1	97.28 ± 0.14
800×1	97.37 ± 0.16
100×1	97.89 ± 0.12
120×1	97.76 ± 0.11

Table 5: Results using varying dilation rate in FDCNN architecture. Epochs= 50

d	average testing accuracy $\pm$ standard deviation (%)
1	97.26 ± 0.07
2	98.92 ± 0.05
4	98.78 ± 0.04
8	98.49 ± 0.05
16	97.43 ± 0.05

Table 6: Comparison with state-of the art models on unbalanced (original) ISCX VPN-nonVPN dataset. The results for Deep Packet and DataNet are obtained from [13, 24] respectively.

Architecture	P	R	F1
Deep Packet$_{CNN}$	94	93	93
DtaNet$_{CNN}$	98.47	98.42	98.43
FDCNN	**98.91**	**98.68**	**98.72**

Traffic characterization results on ISCX VPN-nonVPN are shown in Table 2 after a run over 50 epochs. One could see that FDCNN delivered top precision on the ISCX VPN-nonVPN dataset with higher precision, recall and F1-Score. Table 6 shows that the FDCNN method performed considerably well over baselines with an average improvement of 0.40% compared to that of models trained on adversarial samples. Summarizing, the main reason for

FDCNN architecture performing better in comparison to SH architecture is due to the fact that the former has more information to make a decision.

We also compared the time occupancy performance of FDCNN with the SH architecture in Table 3. By comparison, it can be seen that the FDCNN operation consumes more time especially on a balanced ISCX VPN-nonVPN dataset. Considering that GPUs can greatly improve the computational efficiency of FDCNN and the continuous improvement of computing power in recent years, FDCNN still has great advantages.

4.3 Effect of Hyperparameters on FDCNN Architecture

The filter size k and dilation factor d are the parameters playing an important role in FDCNN architecture. For this reason, in this experiment we analyze the performance of the FDCNN architecture as both parameter sizes vary, providing a more convincing and comprehensive understanding of the proposed architecture. The effect of different hyperparameters settings are displayed in Table 4 and 5 respectively.

In Table 4, the average testing accuracy increases at a rate 98.35% on the ISCX VPN-nonVPN dataset with filter size 20 and then decreases as the filter size increase. For ISCX VPN-nonVPN dataset with kernel size $k > 20$ results in overfitting and the decrease in the generalization ability of the FDCNN model since there is an increase in model parameters.

We also investigate the effect of different dilatation rates in each head of FDCNN architecture. We set the lower layer Convolutions dilation rate to a constant 2 and 1, an advice from [28], then we set the d in all heads with the same dilation rate in the range { 1, 2, 4, 8, 16}. Furthermore, we also investigate the overall average performance of FDCNN with different d's in each head. The results are shown in In Table 5.

In Table 5, for dilatation rate of 1 in all heads, the FDCNN convolutions are conventional Convolutional neural networks, the average testing accuracy is 97.99%. For $d > 1$, we could see that the testing accuracy rises and fall considerable at $d > 4$ on both datasets. For different d's in each head, the average testing accuracy remains higher on both datasets. In conclusion, using a fusion dilated Convolutional neural network with different dilated rates and filter sizes is more effective in supporting exponential large receptive field and also capture local dependencies in the input data than using a conventional Convolutional counterpart. We also conclude that setting dilation rates to 2 and 4 with filter sizes of 20 and 80 in FDCNN architecture will result in optimal performance.

5 Conclusion

In this paper, we present FDCNN architecture for characterizing encrypted web traffic. The FDCNN learns features in encrypted web traffic generated by each application without the need to decrypt the packets in order to classify them, rather, extract and learn from those discriminative patterns so doing it can classify encrypted traffic accurately. Our proposed model integrates fusion dilated Convolutional network with an Attention mechanism to learn range dependencies from web traffic data.

We conducted extensive experiments on standard dataset ISCX VPNnonVPN to demonstrate the effectiveness of our proposed architecture for the problem of encrypted web traffic characterization, further, we also evaluated the characterization abilities of both architectures on anomalous web traffic data. The empirical results from both experiments show that the proposed method outperforms the Single-head architecture by a large margin.

In the future work, we intend to focus on analyzing the ability of FDCNN to adapt to configurations such as when new heads are installed, removed, or modified.

References

[1] V. Aghaei-Foroushani, A. N. Zincir-Heywood, "A proxy identifier based on patterns in traffic flows," in *IEEE 16th International Symposium on High Assurance Systems Engineering*, pp. 118-125, 2015.

[2] S. Bagui, X. Fang, E. Kalaimannan, S. C. Bagui, J. Sheehan, "Comparison of machine-learning algorithms for classification of vpn network traffic flow using time-related features," *Journal of Cyber Security Technology*, vol. 1, no. 2, pp. 108-126, 2017.

[3] S. Bai, J. Z. Kolter, V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," arXiv preprint arXiv:1803.01271, 2018.

[4] A. Dainotti, A. Pescape, and K. C. Clay, "Issues and future directions in traffic classification," *IEEE Network*, vol. 26, no. 1, 2012.

[5] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, A. A. Ghorbani, "Characterization of encrypted and vpn traffic using time-related features," in *Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP'16)*, pp 407-414, 2016.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.

[7] E. Hodo, X. Bellekens, E. Iorkyase, A. Hamilton, C. Tachtatzis, R. Atkinson, "Machine learning approach for detection of nontor traffic," in *Proceedings of the 12th International Conference on Availability, Reliability and Security*, pp. 1-6, 2017.

[8] J. Khalife, A. Hajjar, and J. Diaz-Verdejo, "A multilevel taxonomy and requirements for an optimal traffic-classification model," *International Journal of Network Management*, vol. 24, no. 2, pp. 101-120, 2014.

[9] D. P. Kingma, J. Ba, "ADAM: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS'12)*, pp. 1097-1105, 2012.

[11] M. Korczynski, A. Duda, "Classifying service flows in the encrypted Skype traffic," in *IEEE International Conference on Communications (ICC'12)*, pp. 1064-1068, 2012.

[12] A. H. Lashkari, G. Draper-Gil, M. S. I. Mamun, A. A. Ghorbani, "Characterization of Tor traffic using time based features," in *Proceeding of the 3rd International Conference on Information System Security and Privacy*, pp. 253-262, 2017.

[13] M. Lotfollahi, M. Jafari Siavoshani, R. Shirali Hossein Zade, M. Saberian, "Deep packet: A novel approach for encrypted traffic classification using deep learning," *Soft Computing*, vol. 24, no. 3, pp. 1999-2012, 2020.

[14] V. Mnih, N. Heess, A. Graves, "Recurrent models of visual attention," *Advances in neural information processing systems*, vol. 27, pp. 2204-2212, 2014.

[15] V. Nair, G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010.

[16] Y. Qi, L. Xu, B. Yang, Y. Xue, and J. Li, "Packet classification algorithms: From theory to practice," in *IEEE INFOCOM*, pp 648-656, 2009.

[17] T. Salimans, D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," *Advances in neural information processing systems*, vol. 29, 2016.

[18] S. Sen, O. Spatscheck, and D. Wang, "Accurate, scalable in-network identification of p2p traffic using application signatures," in *Proceedings of the 13th International Conference on World Wide Web*, ACM, pp 512-521, 2004.

[19] . Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *4th International Conference on Information Systems Security and Privacy (ICISSP'18)*, pp. 108-116, 2018.

[20] J. Sherry, C. Lan, R. A. Popa, and S. Ratnasamy, "Blindbox: Deep packet inspection over encrypted traffic," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pp. 213-226, 2015.

[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929-1958, 2014.

[22] C. Suman, S. Tripathy, S. Saha, "Building an effective intrusion detection system using unsupervised feature selection in multi-objective optimization framework," *arXiv preprint arXiv:1905.06562*, 2019.

[23] H. Ting, W. Yong, and T. Xiaoling, "Network traffic classification based on kernel self-organizing maps," in *International Conference on Intelligent Computing and Integrated Systems*, pp 310-314, 2010.

[24] 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.

[25] W. Wang, M. Zhu, J. Wang, X. Zeng, Z. Yang, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," in *International Conference on Intelligence and Security Informatics*, pp 43-48, 2017.

[26] Z. Wang, "The applications of deep learning on traffic identiffication," *BlackHat USA*, vol. 24, no. 11, pp. 1-10, 2015.

[27] I. K. Yeo and R.A. Johnson, "A new family of power transformations to improve normality or symmetry," *Biometrika*, vol. 87, no. 4, pp. 954-959, 2000.

[28] F. Yu, V. Koltun, T. Funkhouser, "Dilated residual networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 472-480, 2017.

[29] M. Zeng, L. T. Nguyen, B. Yu, O. J. Mengshoel, J. Zhu, P. Wu, J. Zhang, "Convolutional neural networks for human activity recognition using mobile sensors," in *6th International Conference on Mobile Computing, Applications and Services (MobiCASE'14)*, pp. 197-205, 2014.

Biography

Benjamin Appiah is currently a lecturer at Ho Technical University. His research interests include machine learning, Security, Deep Learning, Data Mining, and Big Data Analysis.

Anthony Kingsley Sackey received his M.Sc. degree from Kwame Nkrumah University of Science and Technology, Ghana, He is currently an Assistant lecturer at Ho Technical University. His research interests include Machine Learning, Educational Technology, Human Computer Interaction and Learning Analytics.

Owusu-Agyemang Kwabena received the M.Sc. degree from Coventry University and Ph.D. degree from University of Electronic Science and Technology of China. His research interests include machine learning, data mining, big data analysis, applied cryptography, blockchain technology, and medical image processing.

Ansuura JohnBosco Aristotle Kanpogninge is currently a Ph.D. candidate at CK Tedam University Ghana. His research interests includes Cryptography, Block Chain, data mining, and big data analysis.

Peter Antwi Buah received the M.Sc. degree from University of Electronic Science and Technology of China. He

is currently pursuing the Ph.D. degree at Southwest Jiaotong University. His research interests include landslides, slopes and earthquake.

Research on the Security Protection of Network Communication Data Using DES Encryption Algorithm

Tao Wang[1] and Jia Wang[2]
(Corresponding author: Tao Wang)
Information Technology Center, Hebei Finance University, Baoding, Hebei 071000, China[1]
No. 3188, Hengxiang North Street, Baoding, Hebei 071000, China
Email: ttia15@163.com
Experimental Teaching Center, Hebei Finance University, Baoding, Hebei 071000, China[2]
(Received May 30, 2020; Revised and Accepted May 28, 2022; First Online June 23, 2022)

Abstract

This paper briefly introduced the basic principle and encryption steps of the data encryption standard (DES) algorithm in network communication. Then, it improved the DES algorithm with advanced encryption standards (AES) to form a DES+AES hybrid encryption algorithm. Next, the hybrid encryption algorithm was tested for plain text and key sensitivity. Finally, in order to verify the performance of the hybrid encryption algorithm, the hybrid algorithm was compared with AES and DES algorithms through experiments. The final results showed that the hybrid encryption algorithm had good plaintext and key sensitivity. In addition, it had higher encryption efficiency and stronger resistance to third-party cracking than the other two encryption algorithms.

Keywords: Advanced Encryption Standard; Data Encryption Standard; Encryption Algorithm; Network Communication

1 Introduction

The birth and development of the Internet has enriched people's daily life and improved the productivity of society, but efficient network information transmission also brings inevitable network security problems at the same time [8]. The more people rely on the Internet in their daily production life, the more the private and sensitive information on the Internet. Private information faces the possibility of being intercepted by hackers in the transmission process and being used in illegal acts [11]. Therefore, in order to ensure the information security of Internet users, it is necessary to encrypt the information during the data transmission [13]. Guo *et al.* [4] proposed an image encryption scheme based on fractional-order chaotic time series. The experimental results showed that the key space was large enough to resist brute force attacks and the gray value distribution of encrypted images had randomness.

Min *et al.* [14] proposed a density-based data encryption scheme and a database outsourcing query processing algorithm. In the performance analysis, compared with the existing scheme, the new scheme had better query processing performance and ensured the user's privacy. Singh *et al.* [12] proposed the dual security layer for data. The first layer encoded the data using the least significant bit image steganography method, and the second layer encrypted the data using the advanced encryption standard (AES) algorithm. The experimental results verified the effectiveness of the method. This paper first briefly introduced the basic principle and encryption steps of the data encryption standard (DES) algorithm in network communication and then improved it with the AES algorithm to form a DES+AES hybrid encryption algorithm. The plain text and key sensitivity of the hybrid encryption algorithm were tested through simulation experiments. The hybrid algorithm was also compared with DES and AES algorithms.

2 DES Algorithm for Network Communication Protection

On the Internet, every user's communication device can be regarded as a communication node, and data are transmitted between nodes and nodes in the network channel with the help of communication protocols [5]. This paper encrypted the data in network communication with the DES encryption algorithm. Figure 1 shows the basic principle of the DES encryption algorithm in network communication [9]. The plain text of user data is encrypted at the sending node using the DES encryption algorithm, and the cipher text is transmitted in the communication network according to Transmission Con-

trol Protocol/Internet Protocol (TCP/IP); the receiving node receives the transmitted cipher text according to TCP/IP [7] and decrypts it using the same key, i.e., the inverse operation of DES encryption is performed on the cipher text.

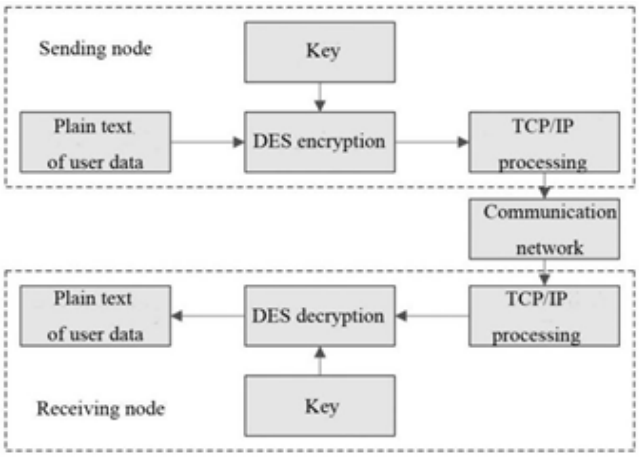


Figure 1: Basic principle of DES encryption in network communication

Figure 2 shows the basic process of the DES encryption algorithm [10]. The specific encryption steps for every group of plain text are as follows.

1) The initial permutation (IP) is performed on 64-bit plain text, i.e., the original order of the 64 characters is disrupted according to some rules. The rules of IP are fixed [3].

2) The disordered 64-bit plain text is divided into two groups of strings, the left string and the right string. Every set was 32-bit long. The loop iteration count i is denoted as 0.

3) The right 32-bit string is used as input data and converted to 48-bit data according to the expansion rule. A 48-bit subkey is generated through the round key generator to perform exclusive OR (xor) operation on the expanded data [1].

4) The 48-bit data encrypted with the key is divided into 8 groups of 6-bit data, and every group of data is subjected to S-box replacement operation. The number of rows in the S-box table of every group depends on the 1st and 6th bit data, and the number of columns in the S-box table depends on the 2nd, 3rd, 4th and 5th bit data. After the S-box substitution operation, eight groups of 4-bit data are obtained, and 32-bit data are obtained after merging.

5) The xor operation is performed on the 32-bit data obtained after the S-box substitution operation and the left 32-bit input data to obtain the right 32-bit output data. The loop iteration count i is added by 1.

6) Whether the loop iteration count i reaches 16 is determined. If not, the loop continues, the right 32-bit input data are used as the left 32-bit input data for the next loop, the right 32-bit output data are used as the right 32-bit input data for the next loop, and it finally returns to Step 3 for the next loop. If the loop iteration count i reaches 16, i.e., 16 loop iterations are completed, the left 32-bit input data and right 32-bit output data are combined into 64-bit data, and then the transformation process is performed using the inverse IP-1 to obtain the 64-bit cipher text.

3 Improvements to the DES Encryption Algorithm

The traditional DES encryption algorithm has been introduced in the previous section. As a symmetric encryption algorithm, it has a high encryption speed because of the short key length, but the security is low [6]. A 56-bit key is not short, but with the improvement of computer performance, the possibility of using the exhaustive method to decode the 56-bit key violently is also increased. In addition, the same key is used for encryption and decryption in the symmetric encryption algorithm, which further increases the risk.

The shortcoming of the traditional DES encryption algorithm is that the key length is so short that it is easier to be decoded by exhaustive method, i.e., the key security is low. The increase in key length can improve encryption security. Taking the triple encryption of plain text for an example, i.e., perform DES encryption thrice on the plain text with three keys, the key length can be increased to 168 bits; moreover, it can be mixed with the other encryption algorithms for encryption.

For the sake of encryption efficiency, the traditional DES encryption algorithm is improved by the AES encryption algorithm [2]. The improvement method is combining the two algorithms. The AES encryption algorithm is also a symmetric encryption algorithm. 128-bit, 192-bit and 256-bit keys are available for AES encryption. As the number of key bits increases, the time required for encryption and decryption will also increase. Considering that a 128-bit key is enough, it is usually used as the key for AES. The encryption process is shown in Figure 3.

1) The plain text M to be encrypted is divided into two plain texts, M_1 and M_2, according to the character lengths that can be handled by the two encryption algorithms.

2) M_1 is encrypted using the DES encryption algorithm. The detailed steps are the same as the traditional steps described previously.

3) M_2 is encrypted using the AES encryption algorithm along with the DES algorithm [15].

4) C_1 obtained after encrypting M_1 with the DES algorithm and C_2 obtained after encrypting M_2 with

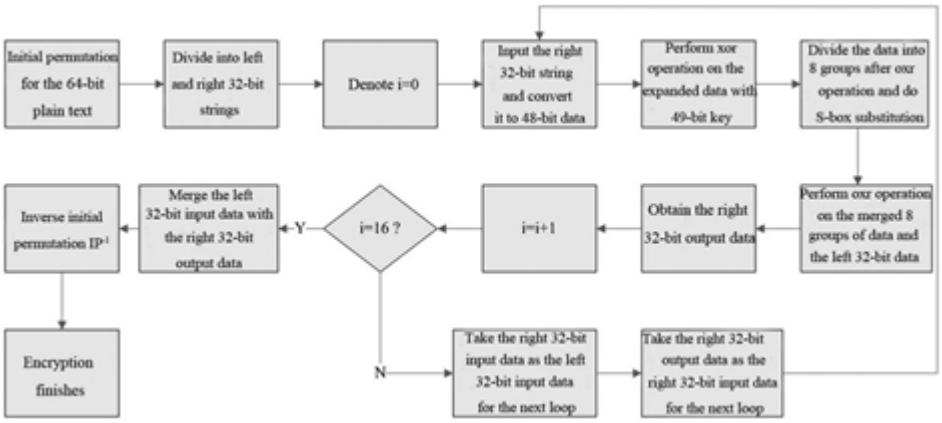


Figure 2: The basic flow of the DES encryption algorithm

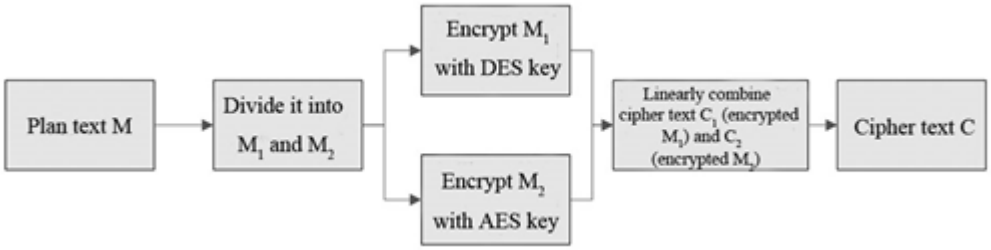


Figure 3: Linear hybrid encryption process of DES+AES

the AES algorithm are combined linearly to obtain cipher text C.

4 Simulation Experiments

4.1 Experimental Environment

The simulation experiments were conducted in the local area network (LAN) of the laboratory. Three servers were used in the simulation experiments. Server 1 was used as a message sending node, server 2 as a message receiving node, and server 3 as a third-party server to act like a hacker to intercept and decrypt the transmitted messages. The communication protocol in the LAN was TCP/IP. The server had 16 G memory, a Core I7 processor, and a 64-bit Windows operating system.

4.2 Experimental Projects

1) Plain text and key sensitivity test for the DES+AES hybrid encryption algorithm
When testing the plain text sensitivity of the hybrid encryption algorithm, three plain texts, "123456abcdef", "123450abcdef" and "123456abcdeg", were given. All three plain texts consisted of both numbers and letters. There was only a difference of one number between the first and second plain texts and only a difference of one letter between the first and third plain texts. The three plain texts were encrypted using the hybrid encryption algorithm with the same key, and the three cipher texts were compared.

The key is divided into AES and DES keys in the key sensitivity test. This paper provides three groups of keys. Only one character of the DES key was different between the first and second groups of keys, and only one character of the AES key was different between the first and third groups of keys. The hybrid encryption algorithm encrypted plain text "123456abcdef" with the three groups of keys, and the three cipher texts were compared.

2) Encryption efficiency test for the DES+AES hybrid encryption algorithm
5 MB, 15 MB, 25 MB, 35 MB and 45 MB packets, 100 each, were set up and transmitted from server 1 to server 2. They were encrypted using the DES+AES hybrid encryption algorithm in the transmission process, and the encryption time was detected. Moreover, the encryption time of using DES and AES algorithms separately was also detected for comparisons.

3) Attack resistance performance of the DES+AES hybrid encryption algorithm
Similarly, 5 MB, 15 MB, 25 MB, 35 MB and 45 MB packets, 100 each, were encrypted and transmitted from server 1 to server 2, passing through server 3; server 3 decrypted the encrypted packets to simulate

the situation that the encrypted data were attacked by the third party during the network communication. The time limit for the third-party server to crack the encrypted data was 60 min, and the decryption integrity was obtained by comparing the cracked cipher text with the original text. The encryption algorithms used in the above experiment were DES, AES and DES+AES encryption algorithms. The attack resistance performance of the three encryption algorithms was compared.

4.3 Experimental Results

Before comparing the hybrid algorithm with AES and DES algorithms, the sensitivity test of the plain text and key was conducted. Figure 4 shows a screenshot of one of the plain texts during hybrid encryption, and the final test results are shown in Tables 1 and 2. Table 1 shows the results of the plain text sensitivity test of the hybrid encryption algorithm. In the plain text sensitivity test, the hybrid encryption algorithm encrypted three plain texts with the same key. There was only a difference of one number between the first and second plain texts, but the encrypted cipher texts had significant changes in the first half part; there was only a difference of one letter between the first and third plain texts, but the encrypted cipher texts had significant changes in the latter part and length.

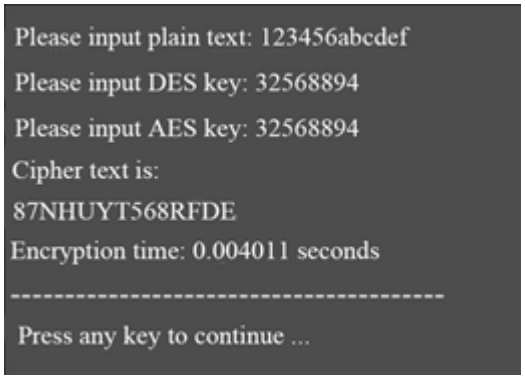


Figure 4: A screenshot of DES+AES encryption

Table 2 shows the test results of the key sensitivity of the hybrid encryption algorithm. In this test, the hybrid encryption algorithm used three groups of keys to encrypt the same plain text. There was only a difference of one character in the key of the AES part between the first and second groups of keys, but the encrypted cipher texts had significant changes in the latter part. There was only a difference of one character in the key of the DES part between the first and third groups of key, but the encrypted cipher texts had significant changes in the first half part.

Encrypting communication data in network communication can improve network security, but it takes time to encrypt the data, so the time consumed to encrypt the data should not be too much in order to guarantee the efficiency of network communication. Figure 5 shows the encryption time required by the three encryption algorithms for different sizes of data packets. It was seen from Figure 5 that the time consumed by the three encryption algorithms increased with the increase of the packet size. The reason for the increased time is that the increased packet size meant the increased number of characters to be encrypted and the number of characters that could be encrypted in a single time was limited. Moreover, when encrypting packets of the same size, the encryption time of AES and DES encryption algorithms were close, but the encryption time of the DES +AES hybrid encryption algorithm was significantly shorter than the other two encryption algorithm. The reason for the shorter time is that the hybrid encryption algorithm split the plain text into two parts and encrypted them with DES and AES algorithms, respectively, which was equivalent to encrypting data that was shorter than the original length with two encryption algorithms.

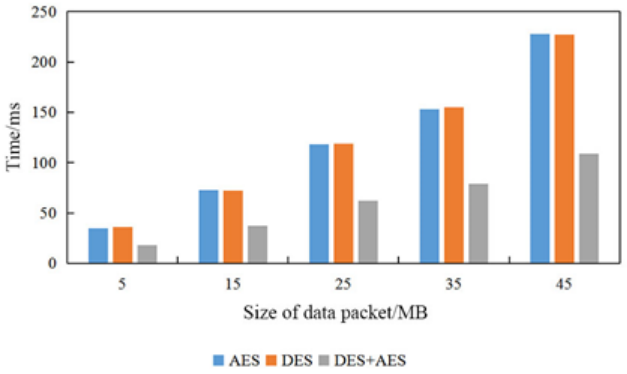


Figure 5: Encryption time

In the process of network communication, it is likely to be attacked by third-party lawbreakers. Although the data in the communication network has been encrypted, there is also the possibility of being cracked, so the resistance of encryption algorithms to attack is also very important. Figure 6 shows the plain text integrity of encrypted packets of different sizes after being cracked by the third party for 60 min. It was seen from Figure 6 that as the packet size increased, the integrity of the plain text encrypted by the three encryption algorithms decreased. The reason is that the increased packet size meant the increased data volume, resulting in increased computation amount during cracking. When facing the packet of the same size, the integrity of the DES-encrypted packet after cracking was the highest, followed by the AES-encrypted packet and the DES+AES-encrypted packet. The reason is that the cipher text was less likely to be cracked by brute force when the length of the key was long.

5 Conclusion

This paper briefly introduced the basic principle and encryption steps of the DES algorithm in network communication and improved the DES algorithm with AES

Table 1: Results of plain text sensitivity test

Plain text	123456abcdef	123450abcdef	123456abcdeg
DES key	32568894	32568894	32568894
AES key	32568894	32568894	32568894
Cipher text	87NHUYT568RFDE	25RTUYN568RFDE	87NHUYT567FGTRQ

Table 2: Results of key sensitivity test

Plain text	123456abcdef	123456abcdef	123456abcdef
DES key	32568894	32568894	32468894
AES key	32568894	42568894	32568894
Cipher text	87NHUYT568RFDE	87NHUYT543RDFE	7856TGFRW568RFDE

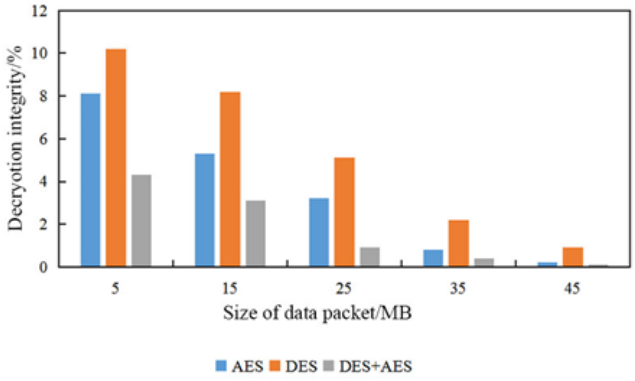


Figure 6: Attack resistance of encryption

to form a DES+AES hybrid encryption algorithm. Moreover, a simulation experiment was performed on the hybrid encryption algorithm. The cipher text and key sensitivity of the hybrid encryption algorithm were tested, and the hybrid encryption algorithm was compared with DES and AES encryption algorithms. The results are as follows. The cipher text encrypted by the hybrid encryption algorithm showed significant changes, as long as there was a change of one character in the cipher text of both plain text and key, indicating that the hybrid encryption algorithm was sensitive to plain text and key. As the packet size increased, the encryption time consumed by all the three encryption algorithms increased; when facing the packet of the same size, the encryption time of AES and DES algorithms was close, and the encryption time of the hybrid encryption algorithm was significantly shorter. As the packet size increased, the integrity of the plain text cracked by the three encryption algorithms decreased; when facing the packet of the same size, the integrity of the DES-encrypted plain text after cracking was the highest, followed by the AES algorithm and the DES+AES hybrid encryption algorithm.

References

[1] S. Ajili, M. A. Hajjaji, A. Mtibaa, "Combining watermarking and encryption algorithm for medical image safe transfer: method based on DCT," *International Journal of Signal and Imaging Systems Engineering*, vol. 9, no. 4/5, pp. 242, 2016.

[2] M. Chen, "Accounting data encryption processing based on data encryption standard algorithm," *Complexity*, vol. 2021, no. 5, pp. 1-12, 2021.

[3] G. Eltayeb, "Combination of the "Data encryption standard" algorithm (DES) and the 'pub-lickey encryption' algorithm (RSA) on the key generation stage," *International Journal of Emerging Trends & Technology in Computer Science*, vol. 4, no. 4, pp. 1-3, 2015.

[4] Z. Guo, J. Yang, Y. Zhao, "Double image multi-encryption algorithm based on fractional chaotic time series," *Open Mathematics*, vol. 13, no. 1, pp. 868-876, 2015.

[5] Y. M. Koukou, S. H. Othman, M. Nkiama, "Comparative study of AES, Blowfish, CAST-128 and DES encryption algorithm," *IOSR Journal of Engineering*, vol. 06, no. 6, pp. 01-07, 2016.

[6] O. Laia, E. M. Zamzami, Sutarman, "Analysis of combination algorithm data encryption standard (DES) and Blum-Blum-Shub (BBS)," *Journal of Physics: Conference Series*, vol. 1898, no. 1, pp. 012017 (7pp), 2021.

[7] Z. L. Lan, L. Zhu, Y. C. Li, J. Liu, "A color image encryption algorithm based on improved DES," *Applied Mechanics & Materials*, vol. 743, pp. 379-384, 2015.

[8] W. H. Liu, K. H. Sun, C. X. Zhu, "A fast image encryption algorithm based on chaotic map," *Optics & Lasers in Engineering*, vol. 84, pp. 26-36, 2016.

[9] Z. Mihret, M. W. Ahmad, "The reverse engineering of reverse encryption algorithm and a systematic comparison to DES," *Procedia Computer Science*, vol. 85, pp. 558-570, 2016.

[10] A. B. Nasution, S. Efendi, S. Suwilo, "Image steganography in securing sound file using arithmetic coding algorithm, triple data encryption standard (3DES) and modified least significant bit (MLSB)," *Journal of Physics Conference*, vol. 1007, pp. 012010-, 2018.

[11] V. C. Rashmi, R. Sehgal, R. Nagpal, "The RC7 encryption algorithm," *International Journal of Security and its Applications*, vol. 9, no. 5, pp. 55-60, 2015.

[12] S. Singh, V. K. Attri, "Dual layer security of data using LSB image steganography method and AES encryption algorithm," *International Journal of Signal Processing Image Processing & Pattern Recognition*, vol. 8, no. 5, pp. 259-266, 2015.

[13] X. Wang, H. L. Zhang, "A novel image encryption algorithm based on genetic recombination and hyperchaotic systems," *Nonlinear Dynamics*, vol. 83, no. 1-2, pp. 333-346, 2016.

[14] M. Yoon, M. Jang, Y. S. Shin, J. W. Chang, "A bitmap based data encryption scheme in cloud computing," *International Journal of Software Engineering & Its Applications*, vol. 9, no. 5, pp. 345-360, 2015.

[15] H. Yue, X. Zheng, "Research on encrypting accounting data using des algorithm under the background of microprocessor system," *Microprocessors and Microsystems*, pp. 104061, 2021.

Biography

Tao Wang, born in 1973, has received the master's degree from Hebei University. He is a professor and is working in Hebei Finance University. He is interested in informatization.

Jia Wang, born in 1987, has received the master's degree from North China Electric Power University in 2013. She is a lecturer and is working Hebei Finance University. She is interested in informatization.

Identification of Traffic Flow Using Multi-Convolutional Neural Networks

Ching-Ta Lu[1,2,3], Yu Huang[1], Jia-An Lin[4], and Ling-Ling Wang[1]
(Corresponding author: Ling-Ling Wang)

Department of Information Communication, Asia University[1]
Taichung City, 41354, Taiwan, R.O.C.
Email: ling@asia.edu.tw
Department of Medical Research, China Medical University Hospital, China Medical University[2]
Taichung City 40402, Taiwan, R.O.C.
Department of Audiology and Speech-Language Pathology, Asia University[3]
Department of Digital Media Design, Asia University[4]
Taichung City 41354, Taiwan, R.O.C.
(Received Jan. 13, 2021; Revised and Accepted May 18, 2022; First Online June 28, 2022)

Abstract

Many countries suffer from traffic jams on a freeway or expressways. Moreover, governments significantly consider treating traffic jam-related problems in most modern cities. Therefore, developing a valid and reliable automatic analysis method for detecting expressway traffic flow for the control of each interchange gateway is essential. This study uses two convolutional neural networks (CNNs) to recognize the expressway's traffic flow. Initially, a road-region CNN is employed to recognize the lane regions of a captured image. At the same time, a traffic-flow CNN is used to identify the traffic flow of the image processing by the road-region CNN. The identified results contain three categories: block, more cars, and smooth. The experimental results reveal that the recognized precision rate of the traffic flow can reach 92.5%. Accordingly, the recognition results can control the number of vehicles entering and expressway in the interchange gateways, preventing traffic jams.

Keywords: Convolutional Neural Network; Deep Learning; Traffic Flow Detection; Traffic Jams

1 Introduction

The problem of highway traffic jams plagues most cities globally, while traffic jams cause air pollution. Many government officials in most countries are committed to improving traffic smoothness to improve people's lives and promote economic development. In Taiwan, many cities have a well-developed transportation network, and high vehicle density leads to traffic jams in the expressway. It is still an essential task to reduce the problem of traffic jams.

Lee and Bovik [11] proposed a method for collecting traffic flow information from urban traffic scenes. Traffic flow is calculated by optical flow estimation. The goal is to collect a macroscopic view of traffic flow information in a fully automatic and segmentation-free way. Pratama *et al.* [23] proposed adaptive traffic lights to control the timing through traffic density calculated from the road pattern. The results reveal that estimated traffic density can well control traffic light's timing to improve traffic jams. Wang *et al.* [29] presented an interactive system for analyzing urban traffic congestion based on GPS trajectories. They extract and derive traffic jam information. Spatial and temporal events are concatenated in traffic jam propagation graphs. These graphs create a description of a traffic jam and its propagation in time and space. Luo *et al.* [18] used a cell transmission model to develop control strategies for the dispersion accident-induced traffic jam. They also utilized Timed Petri nets to model variable traffic light control systems, enabling the traffic jam propagation to be mitigated. Lei *et al.* [12] proposed analyzing the difference in consecutive sampled images to extract the ultimate background. Hence, they use the corner feature to detect traffic jams according to the extracted background image. Abadi *et al.* [1] proposed a dynamic simulator to generate traffic flows. They used an autoregressive model with real-time and estimated traffic data to predict the traffic flows about 30 minutes ahead. Jia and Xing [8] presented an information collection method for dynamical traffic flow. This study also provides a theoretical basis for controlling the speed of a vehicle on the freeway. Xiao and Wang [30] proposed a traffic index cloud map to visualize the traffic states for large road network areas. This study uses a traffic index to evaluate the traffic state. Mohammed and Kianfar [20] proposed

using machine learning for short-term traffic flow prediction. Four categories of predictive methods for traffic flow prediction were investigated: deep neural networks, a distributed random forest, a gradient boosting machine, and a generalized linear model. The results show that using the features including speed, traffic flow, occupancy, and the time of a day can reduce the traffic prediction error. Liu *et al.* [13] proposed using an adaptive neuro-fuzzy inference system for short-term traffic flow prediction. A forecast model is presented based on the dynamic traffic flow information. The correlation between traffic flow and vehicle speed at different driveways is analyzed. The results show that the traffic flows in the same direction in different driveways are correlated significantly. Liu *et al.* [14] proposed using an artificial neural network (ANN) to predict urban traffic flow. The first Lyapunov exponent and recurrence plots are used to analyze the forecasting properties of a traffic flow. The ANN predicts the traffic flow according to the analyzed properties. Dai *et al.* [4] proposed combining spatial and temporal analysis with a gated recurrent unit for short-term traffic flow prediction. The correlations for time and spatial domains are computed and regarded as spatio-temporal features employed to define the time interval and spatial data volume. Hence, the gated recurrent unit predicts the traffic flow according to the spatio-temporal features. Ke *et al.* [9] proposes an analysis framework for traffic flow parameter estimation from unmanned-aerial-vehicle video. At first, an ensemble classifier, including the Haar classifier and convolutional neural (CNN), is utilized for vehicle detection. Hence, traffic flow parameters are estimated according to the optical flow and traffic flow theory. Lan and Chang [10] presented the findings from empirical observations of three-class traffic flows, including the complex interactions among scooters, passenger cars, and buses. By analyzing recorded data, including individual vehicles and their trajectories over time, formulations were proposed to model the behaviors of the three-class flows on queue formation, discharging, lane choice in filtering processes, and propagation.

Deep-learning neural networks (DLNNs) have been progressively developed in image processing [2, 3, 5, 6, 15–17, 19, 21, 22, 24, 26–28]. Sun *et al.* [26] proposed using a CNN to inspect surface flaws based on adaptive multiscale image collection. Chen *et al.* [3] proposed a deep Siamese convolutional multiple-layers recurrent neural network for change detection in multi-temporal very-high-resolution images. This method can be used for homogeneous and heterogeneous images. Moeskops *et al.* [19] presented a method for the automatic segmentation of magnetic-resonance brain images into several tissue classes using a convolutional neural network. Oktay *et al.* [22] proposed a training strategy incorporating prior anatomical knowledge into CNNs through a regularization model. The framework enables a model to follow the structures' global anatomical properties, such as labels and shapes. Tian [27] proposed training the CNN and recurrent neural networks in parallel. A residual module ShortCut3-ResNet is constructed, enabling the CNN to learn the image's various features well. Verma *et al.* [28] utilized a CNN and a DNN for facial emotion recognition. This method can effectively categorize a facial expression into seven different categories: afraid, angry, disgusted, happy, neutral, sad, and surprised. Sapijaszko and Mikhael [24] proposed using a multilayer sigmoid neural network for face recognition. They used a gray scaling algorithm to enhance images, and the salient features are extracted using the coefficients obtained by the wavelet transform and cosine one. Mu *et al.* [21] proposed an unsupervised image segmentation method based on region-combined images using super-pixels with deep learning models. The super-pixel segmentation algorithm segments an image, and then the semi-supervised region is merged into an unsupervised algorithm. Finally, the processed image is input into the deep learning model to obtain segmentation results. Fu *et al.* [6] proposed an image super-resolution approach. They replace the default up-sampling layer with a pixel shuffling layer to speed up the calculation of the DLNN. Dilawari *et al.* [5] proposed using some computer vision algorithms with deep learning to extract the spatio-temporal annotations of humans by bounding boxes. The accuracy rates can reach 95.5%. Appathurai *et al.* [2] proposed using an ANN and oppositional gravitational search optimization method for vehicle detection.

Based on the above discussions, CNN performs well in image recognition and classification. We use road region CNN (RR-CNN) and traffic flow CNN (TF-CNN) to recognize the freeway traffic flow. First, images are periodically captured and converted to gray levels. In turn, the captured image is separated into N regions. Each region is identified whether it is inside the lane region by the RR-CNN. The TF-CNN will further recognize the traffic flow for the regions inside the lane. On the contrary, the TF-CNN does not recognize the traffic flow in the non-lane regions. The recognized traffic flow results include three categories: block, many cars, and smooth. The experimental results show that the recognition accuracy rate is very high, so the research results can be applied to the traffic light control for vehicles entering into the freeway at each interchange to prevent the freeway from traffic jams.

The rest of this paper is organized as follows. Section 2 describes the proposed method of using the RR-CNN and TF-CNN for traffic flow identification. Experimental results are demonstrated in section 3. Conclusions are finally drawn in Section 4.

2 Proposed RR-CNN and TF-CNN for Traffic Flow Estimation

The block diagram of the proposed traffic flow identification system is shown in Figure 1. First, the images captured in the freeway are converted to a gray level. Hence,

the lane lines' boundary is detected by the Canny edge detection algorithm [25], as shown in Figure 2. The image is segmented into N regions, where N is empirically chosen to 25, and input them into the RR-CNN to identify the lane regions. If a segmented region is identified as not inside the lane, it brings no information concerning traffic flow. This effect of this region is ignored. On the contrary, a region recognized inside the lane is input into the TF-CNN to identify the traffic flow result. The traffic flow is defined according to the majority of the identified results of the segmented regions in the image.

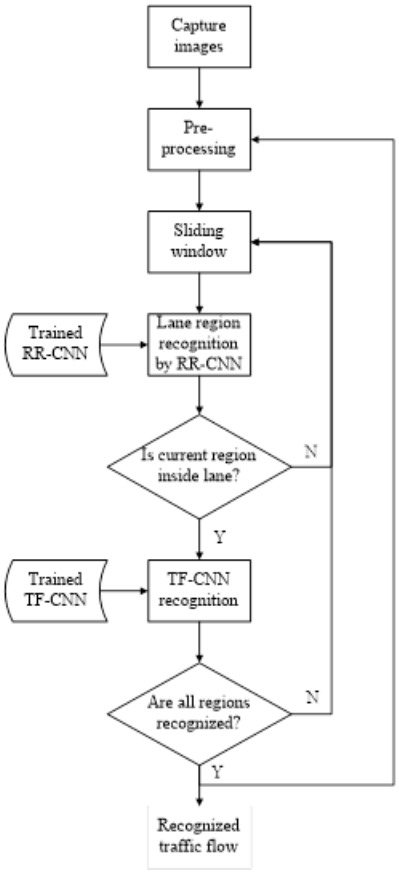


Figure 1: Block diagram of the proposed traffic flow recognition system

(a) (b) (c)

Figure 2: Edge detection using Canny algorithm [25], (a) captured image; (b) grayed image; (c) edges detected by the Canny algorithm

2.1 Pre-processing of Captured Images

To reduce the complexity of identifying the traffic flow, pre-processing such as gray-scale conversion and edge detection is performed on the captured color image. A color image can be converted into grayscale one, the relation is expressed by [7].

$$Y(i,j) = 0.299R(i,j) + 0.58G(i,j) + 0.114B(i,j) \quad (1)$$

where Y represents the gray level. R, G, and B denote the color intensity of red, green, and blue, respectively.

An image captured on the highway is grayed out and the results obtained by Canny edge detection [15] are shown in Figure 2(c). The lane boundary is apparent in the results detected by the Canny algorithm. The detected results are segmented into N equal regions, where N is empirically chosen to 25. The segmented regions are input into the RR-CNN to identify the lane boundary. As shown in Figure 3(c), the RR-CNN can adequately recognize the lane regions. The lane regions recognized by the RR-CNN are input to the TF-CNN to identify the traffic flow.

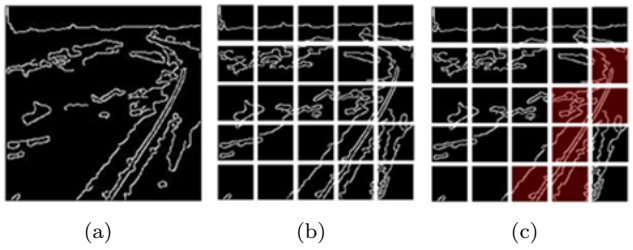
2.2 Traffic Flow Detection by the TF-CNN

According to Google's classification for traffic flow, the traffic flow is divided into three categories, including block, many cars, and smooth. The structure of TF-CNN is shown in Figure 4. The segmented blocks shown in Figure 3 are input into the TF-CNN. The recognized traffic flow is obtained.

Images were utilized for training the TF-CNN. Figure 5 shows the variation of precision rates with various numbers of convolutional layers, which impact the TF-CNN performance. By using three convolutional layer, the best performance in the validation set is achieved. The number of filters for each convolutional layer impacts the performance of TF-CNN. Figure 6 shows the variation of precision rates with various numbers of filters in convolutional layers. Adequately increasing the number of filters can improve the precision rate from 50% to 70%. Using 36 filters in the convolutional layer obtains the best performance. Accordingly, the numbers of convolutional layers and filters are set to 3 and 36, respectively. Figure 7 shows the training trajectory of TF-CNN with three convolutional layers and 36 filters. The accuracy rate of the validation set reaches 92.5

3 Experimental Results

To verify the effectiveness of this paper's method, we collected many images of traffic flow images, where 67% and 37% of them were used as training set and validation set, respectively. The images were all captured from the camera of Taiwan's Highway Bureau of the Ministry of Communications on different dates and times for analyzing and verifying traffic flow.



(a) (b) (c)

Figure 3: Recognized lane region using RR-CNN; (a) edges detected by the Canny algorithm; (b) segmented blocks; (c) lane regions recognized by the RR-CNN

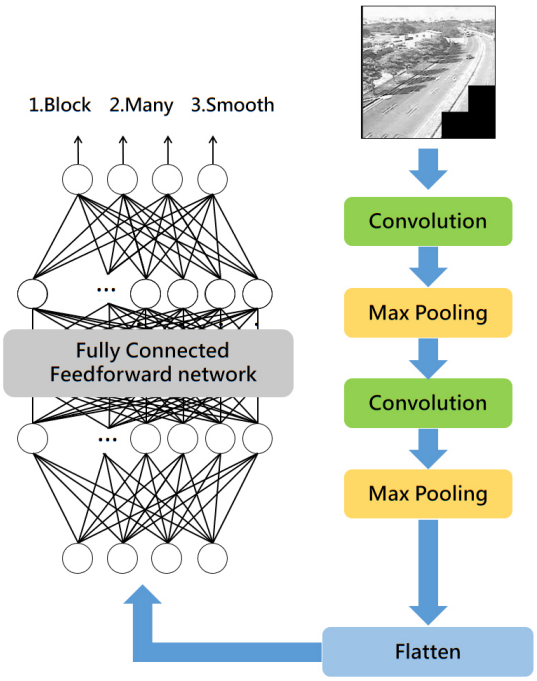


Figure 4: Structure of TF-CNN

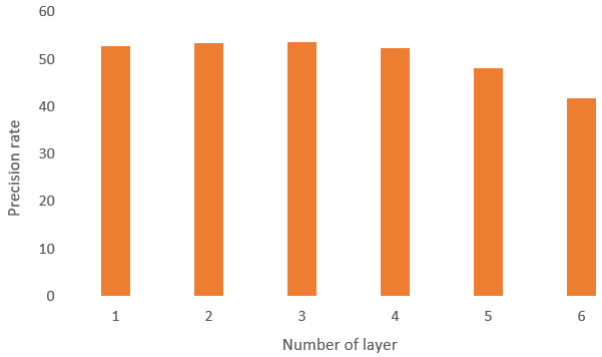


Figure 5: Variation of precision rates with various numbers of convolutional layers

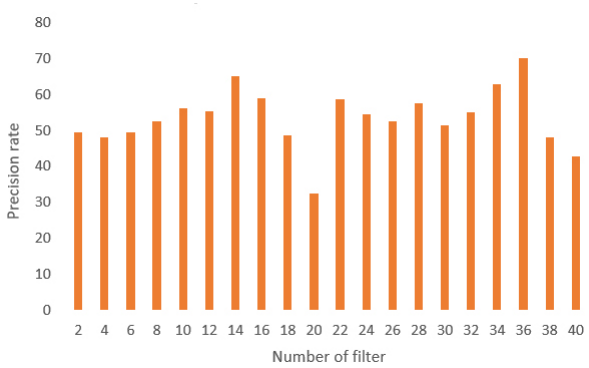


Figure 6: Variation of precision rates with various numbers of filters in convolutional layers

Precision rate can be used to evaluate the correctness of the traffic flow identification, given as

$$P\% = \frac{\text{Number of correctly recognized images}}{\text{Number of images concerning traffic flow}} \cdot 100\% \quad (2)$$

By observing Equation (2), the number of correctly recognized images concerning traffic flow increases, the precision rate $P\%$ improves.

3.1 Lane Boundary Detection

A captured image is segmented into M blocks. The RR-CNN recognizes each block to recognized whether it belongs to the lane region. Figure 8 shows the segmented blocks for M is equal to 25 and 100, respectively. By observing Figure 8(a), the characteristics of the lane boundaries are apparent. However, when an image is divided into 100 blocks, as shown in Figure 8(b), it is hard to see the lane boundary features.

Some training examples for an image having been segmented into 25 blocks and 100 blocks are shown in Figures 9 and **??**, respectively. The validation set's accuracy rate can reach 90% for 100 segmented blocks, whereas the accuracy rate is 92.05% for 25 segmented blocks. Figure 11 shows the performance comparisons in terms of the precision rate. Segmenting an image into 25 blocks performs much better than segmenting into 100 blocks, particularly for the testing set. The primary reason is the less information of lanes for a smaller segmented block, as shown in Figure 10(b). Accordingly, an image is segmented into 25 blocks in the experiments.

3.2 Vehicle Flow Identification

Some permissible recognition types include recognizing a block or a smooth category as more cars category or recognizing more cars category as block or smooth category. This permission is because the category with the recognition result of more cars category has the fuzzy area of smooth or block category, which is also an acceptable

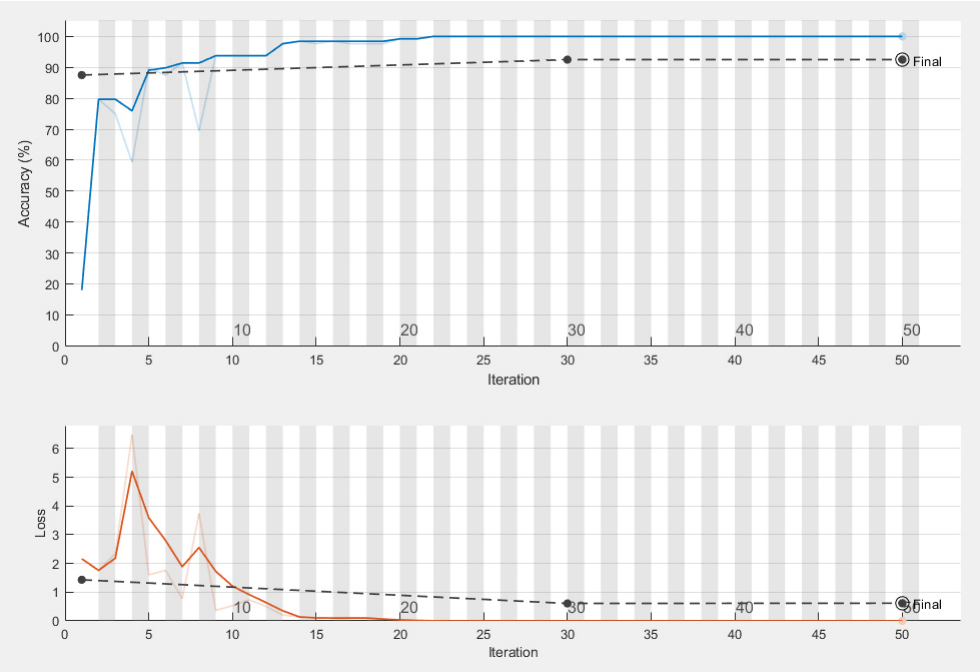


Figure 7: Training trajectory of TF-CNN with 3 convolutional layers and 36 filters; (upper)variation of accuracy rate; (bottom) variation of loss values

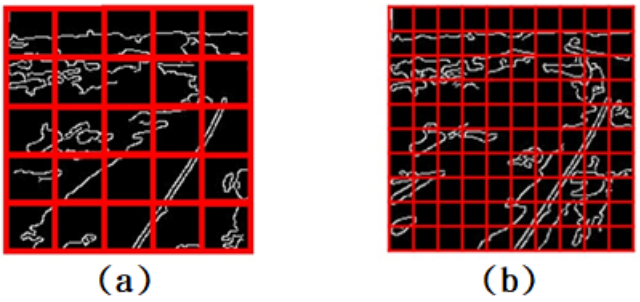


Figure 8: Examples of segmented blocks; (a) 25 blocks; (b) 100 blocks

recognition result, so it is more affordable to list the conditions mentioned above as the allowable recognition category.

Figure 12 shows the precision rates for a captured image been segmented into 25 and 100 blocks. Segmenting an image into 25 blocks is superior to that segmenting the image into 100 blocks. Even the recognized results are not identical to the targets, the recognized results are allowable, i.e., a block target will not be recognized as smooth, and a smooth target will not be recognized as a block. Therefore, the recognized results are acceptable.

Table 1 shows the performance comparisons in precision, error, recall, and allowable recognition ratios. It can be found that the precision rates are identical for each traffic flow conditions. No detection errors have been rec-

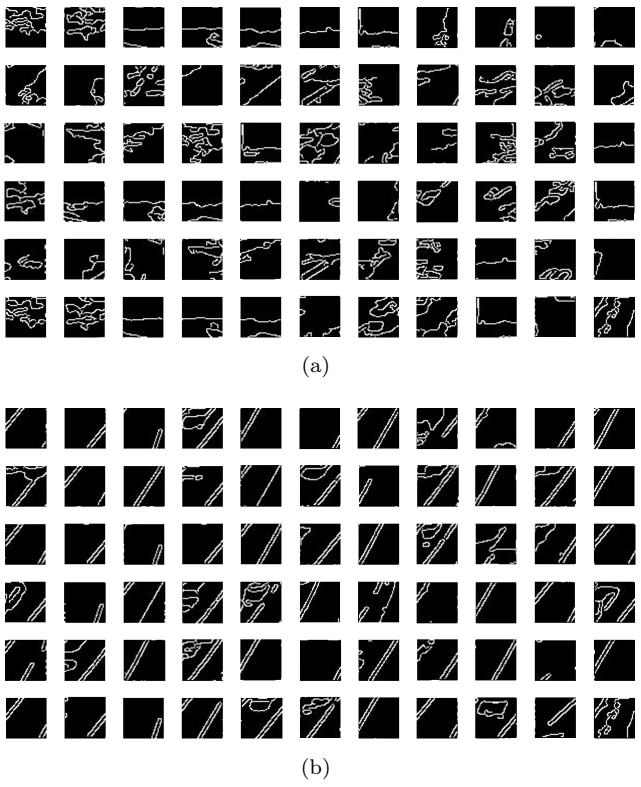
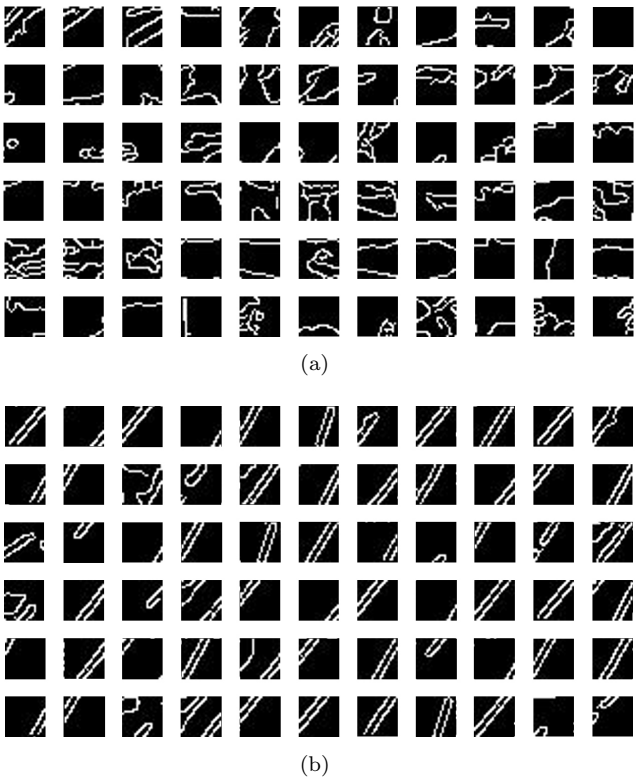


Figure 9: Some examples of training blocks, where an image is segmented into 25 blocks; (a) blocks without lane boundary; (b) blocks with boundary



(a)

(b)

Figure 10: Some examples of training blocks, where an image is segmented into 100 blocks; (a) blocks without lane boundary; (b) blocks with boundary

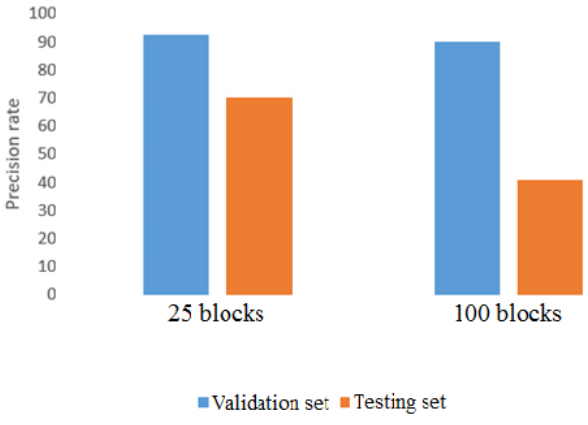


Figure 11: Comparisons of precision rates for images segmented into 25 and 100 blocks

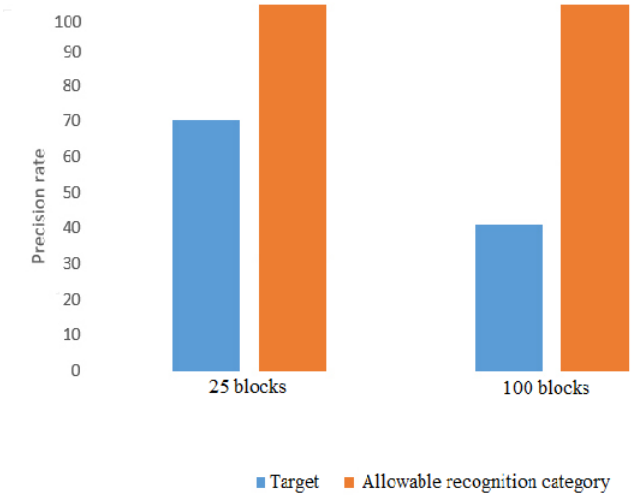


Figure 12: Comparisons of allowable precision rates for images segmented into 25 and 100 blocks

ognized. These results mean that the actual block condition will not be recognized as smooth, and the actual smooth condition will not be recognized as a block. The proposed approach can thoroughly recognize the smooth conditions; thus, the recall rate reaches 100%. The conditions of block and more cars are tough to be distinguished in real environments. The labeled targets may be different obtained by different people. Accordingly, allowable recognition results are acceptable.

Table 1: Performance comparisons in terms of precision, error, recall, and allowable rates

Category	Precision rate (%)	Error rate (%)	Recall rate (%)	Allowable recognition ratio (%)
Smooth	72	0	100	28
More cars	72	0	66.67	28
Block	72	0	74.33	28

4 Conclusions

This paper proposes using two different CNNs to recognize traffic flow. Firstly, the RR-CNN learns to recognize the lanes in each captured image. In turn, the TF-CNN identifies the image's traffic flow in the lane region recognized by the RR-CNN. The precision rate and recall rate evaluate the performance of the traffic flow identification. The experimental results show that the proposed approach can effectively identify traffic flow by using captured images. In the future, this method can be applied to road traffic control in the actual environment.

Acknowledgments

This work was supported by the Ministry of Science and Technology, Taiwan [grant number MOST 104-2221-E-468-007].

References

[1] A. Abadi, T. Rajabioun, P. A. Ioannou, "Traffic flow prediction for road transportation networks with limited traffic data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, pp. 653-662, 2015.

[2] A. Appathurai, R. Sundarasekar, C. Raja, E. J. Alex, C. A. Palagan, A. Nithya, "An efficient optimal neural network-based moving vehicle detection in traffic video surveillance system," *Circuits Syst. Signal Process.*, vol. 39, pp. 734–756, 2020.

[3] H. Chen, C. Wu, B. Du, L. Zhang, L. Wang, "Change detection in multisource VHR images via deep Siamese convolutional multiple-layers recurrent neural network," *IEEE Trans. Geosci. Remote Sensing*, vol. 58, pp. 2848-2864, 2020.

[4] G. Dai, C. Ma, X. Xu, "Short-term traffic flow prediction method for Urban road sections based on space–time analysis and GRU," *IEEE Access*, vol. 7, pp. 143025-143035, 2019.

[5] A. Dilawari, M.U.G. Khan, Z. ur Rehman, K.M. Awan, I. Mehmood, S. Rho, "Toward generating human-centered video annotations," *Circuits Syst. Signal Process.*, vol. 39, pp. 857–883, 2020.

[6] S. Fu, L. Lu, H. Li, W. Wu, A. Paul, G. Jeon, X. Yang, "A real-time super-resolution method based on convolutional neural networks," *Circuits Syst. Signal Process.*, vol. 39, pp. 805–817, 2020.

[7] R. C. Gonzalez, R.E. Woods, *Digital Image Processing*, 3rd ed., New Jersey: Prentice Hall, 2008.

[8] Y. Jia, E. Xing, "The application of traffic flow detection technology on characteristics of freeway traffic flow," in *Proceedings of International Conference on Remote Sens. Environment Transp. Eng.*, pp. 838-840, 2011.

[9] E. Ke, Z. Li, J. Tang, Z. Pan, Y. Wang, "Real-time traffic flow parameter estimation from UAV video based on ensemble classifier and optical flow," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, pp. 54-64, 2019.

[10] C. Lan, G. Chang, "Empirical observations and formulations of Tri-class traffic flow properties for design of traffic signals," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, pp. 830-842, 2019.

[11] J. Lee, A. C., Bovik, "Estimation and analysis of urban traffic flow," in *Proceedings of IEEE International Conference on Image Process.*, pp. 1157-1160, 2009.

[12] X. Lei, W. Qing, C. Xiumin, W. Jun, C. Ping, "Traffic jam detection based on corner feature of background scene in video-based ITS," in *Proceedings of IEEE International Conference on Netw. Sens. Control*, pp. 614-619, 2008.

[13] C. Liu, X. Liu, H. Huang, L. Zhao, "Short-term traffic flow prediction methods and the correlation analysis of vehicle speed and traffic flow," in *Proceedings of International Conference on Comput. Intell. Security*, pp. 415-418, 2008.

[14] J. Liu, H. Fu, X. Liao, "Combination prediction for short-term traffic flow based on artificial neural network," in *Proceedings of World Congress Intell. Control Autom.*, pp. 8659-8663, 2006.

[15] C. T. Lu, R. H. Chen, L. L. Wang, J. A. Lin, "Image enhancement using convolutional neural network to identify similar patterns," *IET Image Process.*, 2020.

[16] C. T. Lu, J. A. Lin, C. Y. Chang, C. H., Liu, L. L. Wang, K. F. Tseng, "Recognition of film type using HSV features on deep-learning neural networks," *J. Electron. Sci. Technology*, vol. 18, pp. 31-41, 2020.

[17] C. T. Lu, L. L. Wang, J. H. Shen, J. A. Lin, "Image enhancement using deep-learning fully connected neural network mean filter," *J. Supercomput.*, 2020.

[18] J. Luo, Y. Huang, Y. Weng, "Design of variable traffic light control systems for preventing two-way grid network traffic jams using timed Petri nets," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, pp. 3117-3127, 2020.

[19] P. Moeskops, M. A. Viergever, A. M. Mendrik, L. S. de Vries, M. J. N. L. Benders, I. Išgum, "Automatic segmentation of MR brain images with a convolutional neural network," *IEEE Trans. Medical Imag.*, vol. 35, pp. 1252-1261, 2016.

[20] O. Mohammed, J. Kianfar, "A machine learning approach to short-term traffic flow prediction: a case study of interstate 64 in Missouri," in *Proceedings of IEEE Int. Smart Cities Conf. (ISC2'18)*, pp. 1-7, 2018.

[21] X. Mu, H. Qi, X. Li, "Automatic segmentation of images with superpixel similarity combined with deep learning," *Circuits Syst. Signal Process.*, vol. 39, pp. 884–899, 2020.

[22] O. Oktay, E. Ferrante, K. Kamnitsas, M. Heinrich, W. Bai, J. Caballero, S. A. Cook, A. de Marvao, T. Dawes, D. P. O'Regan, B. Kainz, B. Glocker, D. Rueckert, "Anatomically constrained neural networks (ACNNs): application to Cardiac image enhancement and segmentation," *IEEE Trans. Medical Imag.*, vol. 37, pp. 384-395, 2018.

[23] B. Pratama, J. Christanto, M. T. Hadyantama, A. Muis, "Adaptive traffic lights through traffic density calculation on road pattern," in *Proceedings of International Conference on Appl. Sci. Technol. (iCAST'18)*, pp.82-86, 2018.

[24] G. M. Sapijaszko, W.B. Mikhael, "Facial recognition system using mixed transform and multilayer sigmoid neural network classifier," *Circuits Syst. Signal Process.*, vol. 39, 2020.

[25] M. Sonka, V. Hilavac, R. Boyle, *Image Processing, Analysis and Machine Vision*, 2nd ed., Boston: PWS publishing, 2008.

[26] J. Sun, P. Wang, Y. Luo, W. Li, "Surface defects detection based on adaptive multiscale image collection and convolutional neural networks," *IEEE Trans. Instrum. Meas.*, vol. 68, pp. 4787-4797, 2019.
[27] Y. Tian, "Artificial intelligence image recognition method based on convolutional neural network algorithm," *IEEE Access*, vol. 8, pp. 125731-125744, 2020.
[28] A. Verma, P. Singh, J. S. R. Alex, "Modified convolutional neural network architecture analysis for facial emotion recognition," in *Proceedings of International Conference on Syst. Signals Image Process. (IWSSIP'19)*, pp. 169-173, 2019.
[29] Z. Wang, M. Lu, X. Yuan, J. Zhang, H. V. D. Wetering, "Visual traffic jam analysis based on trajectory data," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, pp. 2159-2168, 2013.
[30] J. Xiao, H. Wang, "Traffic index cloud maps for traffic flow analysis with big traffic data," in *Proceedings of International Conference on Comput. Commun. Syst. (ICCCS'20)*, pp. 20-23, 2020.

Biography

Ching-Ta Lu received both B.S. and M.S. degrees in Electronic Engineering from National Taiwan University of Science and Technology, Taipei, in 1991 and 1995, respectively; and his Ph. D. degree in Electrical Engineering from National Tsing Hua University, Hsinchu, Taiwan, R.O.C., in 2006. He had been with the Department of Electronic Engineering, Asia-Pacific Institute of Creativity, Miao Li, Taiwan (Aug. 1995- Feb. 2008). He had been the chair of the department in 2000 and 2006. Prof. Lu received the excellent teaching awards in 2006 and 2007, and the best tutor awards in 1986, 1987, 2005, 2006, 2007 at Asia Pacific Institute of Creativity. He also received the excellent teaching awards in 2015 and 2016, and the best tutor awards in 2009, 2013, 2015, 2018, 2020 at Asia University. Currently, he is a full professor of the Department of Information Communication, Asia University (since Feb. 2008). He has published more than 45 journal papers and 70 conference papers. His current research interests include the applications of artificial intelligence, speech enhancement, image recognition, image denoising, speech coding, and speech signal processing.

Yu Huang received B.S. and M.S. degrees in the Department of Information Communication, Asia University, Taiwan, ROC. His research interests include the applications of artificial intelligence and image signal processing.

Jia-An Lin received both B.F.A. and M.F.A. degrees from the Chaoyang University of Technology and Tainan National University of the Arts in 2002 and 2007. Currently, he is an Associate professor of the Department of Digital Media Design, Asia University, Taiwan, R.O.C. He pursuit the Doctoral Program in Art Creation and Theory at Tainan National University of the Arts. His research interests include photographic and video recording practice, digital media industry, documentary production, advertising film production, postcolonial film studies.

Ling-Ling Wang received the B.S., M.S., and Ph.D. degrees in Computer Science and Information Engineering from National Chiao Tung University, Taiwan, ROC in 1984, 1986, and 1990, respectively. From 1986 to 1987, she was an associate engineer in the System Software Department of ERSO, ITRI, Taiwan. From 1991 to 1997, she was an associate professor of Computer Science at National Tsing Hua University, Taiwan. Currently, she is a professor of Information Communication at Asia University, Taiwan. Her major research is in image processing and artificial intelligence.

Chinese Unknown Words Extraction for Incomplete Sentences

Yi-Hui Chen[1,2], Eric Jui-Lin Lu[3], and Jeng-Jie Huang[3]
(Corresponding author: Eric Jui-Lin Lu)
Department of Information Management, Chang Gung University[1]
Taoyuan 33302 Taiwan
Kawasaki Disease Center, Kaohsiung Chang Gung Memorial Hospital[2]
Department of Management Information Systems, National Chung Hsing University[3]
Taichung 402204 Taiwan
Email: jllu@nchu.edu.tw
(Received May 22, 2020; Revised and Accepted Apr. 13, 2022; First Online June 28, 2022)

Abstract

Queried keywords are often used in representing the topics of articles. Word segmentation and unknown word extraction are generally employed to obtain accurate queried keywords. However, existing Chinese unknown word extraction methods are mainly designed to process complete sentences, while the queried keywords are mostly incomplete. In this paper, we propose a Chinese unknown word extraction model for incomplete sentences and use Blog Connect as the experimental platform to collect the queried keywords. A two-phase approach is proposed to solve the unknown word extraction: unknown word detection and unknown word extraction. In the detection phase, we design rules based on the frequency and the probability of queried keywords to detect unknown word candidates. In the extraction phase, we propose a variant of a bottom-up merging algorithm according to pattern and statistical conditions to extract unknown words. The experimental results show that our method can identify about 70% of unknown words and outperforms the CKIP in resolving unknown Chinese words for incomplete sentences.

Keywords: Blog Connect; Queried Keywords; Unknown Words Extraction; Word Segmentation

1 Introduction

As a result of easy access to networks, Web users nowadays conveniently share their experiences and ideas over the Internet [23]. Although Blog readers or bloggers mainly visited their friends' blogs in the past, Technorati's 2011 report [33] pointed out that it is no longer true. Conversely, more and more blog users turn to blog sites that share information they are interested in. Even though blogs on the Internet play an important role in information sharing and dissemination, each blog is still considered as an isolated island [9]. That is, no connections or relationships between any two blogs are established unless they are manually created either by Blogrolls, citation links, or comments [19]. If all related blogs could somehow be "auto-magically" connected, it might result in a breakthrough in information sharing [2]. The past researches [4,22,25] mentioned that a set of keywords of blog posts could be used to represent the main topic for each blog post. Also, the research [5] showed that queried keywords for a blog article could be used to represent the topics of the blog post. Therefore, to achieve high accuracy of keyword extraction is an important issue.

There are a total number of 667 million China Internet users by the end of June 2015. Also, total Internet penetration rate in China reached 48.8% according to CNNIC [15]. The trends showed that Chinese words are widely used to communicate over the Internet. To extract the words syntax in sentences, Chinese word segmentation is an essential preprocessing step. Besides, the segmented results would affect the accuracy of the keyword extraction and its applications such as clustering and classification [29, 30, 39]. However, Chinese language is different from and more challenging than the alphabetic language in that there are no blanks between Chinese words for identifying word boundaries. In the past researches, Chinese queried keywords are segmented by a Chinese segmentation system such as mmseg4j [35] and CKIP [7]. Moreover, Sung *et al.* [34] incorporated multilevel linguistic features to conduct the major tasks of segmentation, syntactic parsing, and feature extraction for analyzing Chinese texts. Lei *et al.* [24] proposed a scheme to detect Chinese unknown words for Weibo tweets and comments. Qiu and Zhang [31] applied the information extraction technique to mine common noun entities for the novels. To extract the unknown words, Ye *et al.* [38] proposed a goodness measure of the candidate to sep-

arate low-frequency meaningful and meaningless words for Chinese sentences effectively. As for evaluating the impact of Chinese word segmentation, Foo and Li [18] used a set of IR experiments to show the influence of different document segmentation approaches on IR effectiveness for query segmentation. However, these Chinese segmentation systems [6, 17, 27, 30] are more suited for a complete sentence, but not quite appropriate for queried keywords because queried keywords are mostly incomplete sentences with words separated by blanks. Thus, queried keywords contain many Chinese unknown words that cannot be identified by existing segmentation systems [4]. That is, a Chinese segmentation system would most likely divide a meaningful queried word into several non-meaningful and separated words. As a result, the original query intensions are lost due to non-meaningful separated words and unidentified unknown words.

The proposed scheme, similar to Chen *et al.*'s [8, 9, 12] and Yang and Chang's [37] studies, extracts unknown words into two phases, namely unknown word detection and unknown words extraction. Based on the frequency and the probability of queried keywords, the proposed scheme designs rules to detect unknown word candidates for incomplete sentences. In the unknown word extraction phase, all morphemes (regardless of whether they are monosyllabic or multisyllabic morphemes) are treated as unknown word morphemes and marked. Then, according to the markups, we build morphological rules for specific types of unknown words, such as the Chinese and English transliteration names. For non-specific types of unknown words, we designed both pattern and statistical conditions to calculate priority values and proposed a variant of the bottom up merging algorithm designed by Chen and Ma [9].

The proposed Chinese unknown words detection and extraction could be applied to search engines, and social platforms (such as Facebook, Twitter, Sina Weibo [32]). In the experiments, the queried keywords of 998 articles collected by the Blog Connect platform are used. Also, we enhanced mmseg4j [35] with our unknown word extraction method, called improved mmseg4j. The experimental results show that there are 299 unidentified unknown words using the improved mmseg4j, while there are 415 unidentified unknown words using CKIP. In other words, our method can extract about 70% (689/988 = 69.7%) of unknown words, but CKIP can only identify less than 58% (573/988 = 57.9%) of unknown words. Therefore, the proposed scheme is more effective than CKIP for incomplete sentences.

2 Related Works

Chen and Lee [10] focus on detecting three types of unknown words, namely Chinese names, English transliteration names, and the names of organizations. In their scheme, Chen and Lee use the following criteria to retrieve the Chinese name: the frequency of characters, sentences, paragraphs, and words; the relationship of context; and the characteristic of the characters. For English transliteration names not distinguishable from a regular pattern, the authors indicated that the translated character used by the English transliteration name is in certain range, such as "Harry Potter", which would be translated in different Chinese terms but with the same pronunciation by different translators, i.e., synonym problem. For this reason, the authors detect the unknown words according to whether or not they are common translation characters and character sequences. As for organization names, the authors believed that an organization name is generally divided into two parts. The first part stands for the name of an organization and the second part denotes the type of the organization. For example, Taichung City Government is divided into Taichung City and Government. In later studies, Chen and Chen [11] used the Academia Sinica Balanced Corpus [1] and a news website to help retrieving organization names and their abbreviated names.

Chen *et al.* [8,9,12] proposed a two-stage method to detect and retrieve unknown words. In the Chen and Bai's scheme [12], it is assumed that every unknown word contains at least one monosyllabic unknown morpheme; thus, rules are trained through the morphological features and their frequencies. If a monosyllabic morpheme applied to the rules, it is considered a known monosyllabic word; otherwise, an unknown one. Based on the Chen and Bai's work [12], Chen and Ma [8] designed the morphological rules to process unknown words such as Chinese names, English transliteration names and so on. In addition, by using the context clues, such as frequency and conditional probability, of the detected unknown words obtained from the previous stage, they defined 12 statistical rules to retrieve unknown words. Chen and Ma [8] demonstrated that an 89% accuracy rate could be obtained in processing 1160 unknown words.

In the past, some research focused on resolving specific types of unknown words. To eliminate such constraints, Chen and Ma [9] presented a bottom up merging algorithm by using a greedy strategy to generate rules for unknown words extraction. However, it is limited by the fact that each unknown word must contain at least one monosyllabic unknown morpheme. When two consecutive terms are all monosyllable unknown morphemes, a statistical value is calculated for each term. Two consecutive terms with the highest statistical value are merged into a token pair. The merging process will continue until no token pairs are unknown.

Some studies use machine learning methods to extract unknown words. Goh *et al.* [20] used a Hidden Markov Model POS Tagger to segment sentences and denote part of speech (POS) tags for each word. After that, the Viterbi algorithm is used to train the corpus to get the optimal sequence of segmented words; then, a SVM chunker is used to detect unknown words. The Peking University corpus is used as the experimental data, and the precision rate and recall rate are 63.8% and 58.3% respectively.

Yang and Chang [37] proposed a hybrid method for

unknown words extraction, in which the unknown words detection is identical to Chen and Bai's scheme [12], while SVM is employed in the unknown words extraction stage. Yang and Chang treated a Chinese sentence as a sequence data, used sliding windows to transform the sequence data into feature vectors, and then utilized classifiers such as SVM to extract unknown words. In the experiments, the precision rate and recall rate are 68% and 63.5%, respectively. Also, Yang and Chang indicated that the F-measure (65.7%) is better than the F-measure (64.8%) of Chen and Ma's scheme [9].

There are studies that used statistical methods to extract unknown words without considering segmentation and parts of speech. Lai and Wu [28] believed that unknown words often have higher co-occurrence rate; thus, they designed a PLU-based likelihood ratio (PLR) and calculated the PLR for each word sequence. For word sequences with the PLR values greater than a threshold, they were treated as unknown words. The experimental results showed that the proposed method had accuracy rate up to 88%. Although Lai and Wu is efficient and fast, Chang and Lee [13] pointed out that unknown words extraction fails when the size of the document set is too small. For small document corpus, Chang and Lee removed stop words from word sequences and proposed a Strict Phrase likelihood ratio (SPLR). After calculating the SPLR for each word sequence, the word sequences with SPLR value greater than a predefined threshold were treated as unknown words. Jiang *et al.* [22] pointed out that 89% of unknown words are generally in the general characteristics. Jiang *et al.* [22] designed both a single character model and an affix mode to extract unknown words. The experimental results showed that accuracy rate up to 87% can be achieved.

3 Proposed Scheme

As stated earlier, the proposed method focuses on extracting unknown words in an incomplete sentence. Therefore, the queried keywords collected by Blog Connect are used as the experimental data set. It differs from other studies that use full-text articles (i.e., complete sentences) as their data sets. In this section, we will describe the Blog Connect first, and then the rest of our proposed scheme.

3.1 Blog Connect

In the past, we developed a platform called Blog Connect (BC) [4] which is a cross-platform system created to discover relationships among blogs with similar interests or topics. Table 1 presents an example of blog post Ai from the BC database. It means that users clicked on Ai four times and the entered keywords are listed in the column "Queried keywords". The column "Number" is simply a number ID for illustration purpose. The column "Meaning of Chinese terms" explains the meaning of queried keywords.

Table 1: Example queried keywords of article A_i

Number	Queried keywords	Meaning of Chinese terms
No.1	羅曼蒂卡 特別	"羅曼蒂卡" (Romantica) is one of unt's model name for nail polish and "特別" means special
No.2	Unt 羅曼蒂卡	Unt is a brand name
No. 3	羅曼蒂卡	"羅" (Pronounced in English is Lo)

In general, a Chinese article is composed of paragraphs, a paragraph is composed of sentences, and a sentence is composed of characters. However, as shown in Table 1, different from complete sentences, queried keywords are strings of characters possibly with blanks to separate words. Thus, the first step is to separate queried keywords into words with blanks. Additionally, all English letters are converted into lower case. It is common that informal words are used for queries. These informal queried keywords are roughly categorized into two categories. The first category contains queried words that have both English and Chinese letters, such as No. 2 in Table 1. The frequency of a queried keyword in all queries is calculated as shown in Table 2. The queried keywords after preprocessing for blog article A_i are denoted as QK_i, where $QK_i = \{k_0, k_1, \cdots, k_n\}$ and n is the number of keywords in QK_i.

Table 2: Example keywords for article A_i after preprocessing

Keyword Number	Queried keyword k	Number in Table 1	frequency
k_0	羅曼蒂卡	No. 3	2
k_1	unt 羅曼蒂卡	No. 2	1
k_2	特別	No. 1	1

3.2 Unknown Words Detection

In the previous studies [12, 37] authors used frequency, part of speech, and conditional probability to train context rules. The context rules are used to determine whether or not a monosyllable morpheme is an unknown word morpheme. However, these methods are limited in that an unknown word morpheme has to contain at least one monosyllable morpheme. Unfortunately, this is not the case for many unknown words. For example, after Chinese segmentation, the queried keyword of "土地銀行" (Land Bank, a short name for Taiwan Land Bank) is split into Land and Bank and both of them are not monosyllable morpheme. Therefore, based on the previous studies, the queried keyword of "Land Bank" is not an unknown word (which is not in any lexicon), but two separated known words (or name entities) that are unrelated to Taiwan Land Bank.

To resolve the limitation, the proposed method utilizes the frequency and conditional probability of k_j in QK_i to detect candidate unknown word. If k_j satisfies the rules in Table 3, k_j is a candidate unknown word and denoted

as UWC_i (Unknown Word Candidate).

Take the rule DR_1 in Table 5 as an example, if k_j satisfies both conditions C_1 and C_2, it is a candidate unknown word. All five conditions used in determining whether or not k_j is a candiate unknown word are summarized in Table 4, where $|k_j|$ is the length of k_j, $f(k_j)$ is the frequency of k_j, and LEX is the word set which is the lexicon of a Chinese segmentation system. Condition C_1 means that k_j is not in the LEX. Condition C_2 is designed by Li *et al.* [26], in which the length of an unknown word is usually between two and four characters. If k_j and k_q are two different keywords in QK_i and k_q is a substring of k_j, there is high probability that k_j is an unknown word. Therefore, condition C_3 is used to detect whether k_j contains a substring k_q and k_q is not in the LEX. If the number of k_j in QK_i is one, it is hard to determine whether or not k_j is an unknown word. As a result, condition C_4 is designed to demand that the number of k_j in QK_i has to be at least 2. Condition C_5 requires that the frequency of a candidate unknown word has to be at least 2.

Table 3: Conditions of unknown word detection

ID	Conditions
C_1	$k_j \notin LEX$
C_2	$\|k_j\| \geq 2$ and $\|k_j\| \leq 4$
C_3	$k_q \notin LEX$ and k_q is a substring of k_j, $1 \leq q \neq j \leq n$
C_4	$n \geq 2$
C_5	$f(k_j) \geq 2$

Table 4: Rules of unknown word detection

Detection Rule ID	Conditions Constraints
DR_1	C_1, C_2
DR_2	C_1, C_3
DR_3	C_1, C_4, C_5

Let us use Table 2 as an example where $QK_i = \{k_0, k_1, k_2\}$. Keyword k_2 is in the LEX, it does not satisfy any rule so that k_2 is a known word. Keyword k_0 is not in the LEX and its length is 4 (which is between 2 and 4), it is a candidate unknown word based on the rule DR_1. In this case, k_0 is put into the UWC_i. Because k_0 is a substring of k_1, and k_1 is not in the LEX, k_1 is also a candidate unknown word based on the rule DR_2. Finally, two keywords k_0 and k_1 "羅曼蒂卡" and "unt酗羅曼蒂卡" are in UWC_i; that is, $UWC_i = \{k_0, k_1\}\{\backslash a", \backslash unta"\}$.

3.3 Unknown Words Extraction

After the unknown word detection stage, the task for the unknown word extraction stage is to extract unknown words from UWC_i. First of all, each k_j in UWC_i is segmented. After segmentation $k_j = \{w_0^p, w_1^p, \cdots, w_c^p, \cdots, w_m^p\}$, where m is the total number of morphemes, and p is a tag variable. If the length of w_c is equal to 1, w_c is tagged as w_c^1 and is a monosyllable morpheme. If the length of w_c is greater than 1, p is equal to the length of w_c, and w_c^p represents a polysyllable morpheme. If w_c is composed of English letters, p is equal to *. Use Table 2 as an example. After segmentation, $k_1 = \{w_0^*, w_1^1, w_2^1, w_3^1, w_4^1, w_5^1\}$ because w_0 is "unt"; and w_1, w_2, w_3, w_4, and w_5 are all of length 1. Noted that w_1, w_2, w_3, w_4, and w_5 are monosyllable morphemes; respectively. Due to the queried keywords collected by Blog Connect are incomplete sentences, the methods proposed in [3, 14, 16, 19] cannot be used. The tasks to extract unknown words from incomplete sentences are proposed and described as follows.

3.3.1 Identification of Chinese Personal Names

In general, a Chinese personal name is composed of a surname and a given name. Most Chinese surnames are using a specific of single character and some rare ones are two characters, such as Chen, Lin, Huang, etc. For clarity, several example Chinese names and their corresponding English transliteration names are listed in Table 6.

In this paper, we use the top one hundred surnames and the common surnames of two characters that are listed in Wikipedia to establish Last Name Set (LN) as our reference. In the stage of Chinese personal name extraction, if w_c^p is contained in LN and p is equal to either 1 or 2 (i.e., w_c^1 or w_c^2), we use rules shown in Table 7 to extract the Chinese personal name. In CR_1, if w_c^1 is in LN and its next two morphemes are w_{c+1}^1 and w_{c+2}^1; w_c^1, w_{c+1}^1 and w_{c+2}^1 are combined as a Chinese personal name. In CR_2, if w_c^1 is in LN, and if the p of w_{c+2}^p is not equal to 1, w_c^1 and w_{c+1}^1 are combined as a Chinese personal name. If w_c^1 is in LN and the p of w_{c+1}^p is equal to 2, then w_c^1 and w_{c+1}^2 are combined as a Chinese personal name based on CR_3. There are two cases considered in CR_4, and the procedure for CR_4 is shown in Figure 1. Firstly, if w_c^2 is in LN (i.e. w_c^2 is a surname of two characters), w_c^2 and w_{c+1}^1 are combined as a Chinese personal name. Secondly, a Chinese segmentation system sometimes segments a Chinese personal name into a disyllable morpheme and a monosyllable morpheme. For example, if k_j is "Chang Ta-Chun" (as No. 1 in Table 6), k_j is split into "Chang-Ta" and "Chun" after segmentation. Therefore, if the first character of w_c^2 is in LN, w_c^2 and w_{c+1}^1 are also combined as a Chinese personal name.

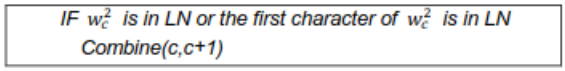


Figure 1: The constraint of No.4 rule type

Table 5: Name example

Keyword Number	Queried keyword	Meaning of Chinese terms	Chinese name translated in English
No. 1	張大春	the name of a notable Taiwanese writer	Chang Ta-Chun
No. 2	布魯斯威利	an American actor	Walter Bruce Willis

Table 6: Rules for Chinese personal names

Rule ID	Rule Type	Constraint & Procedure
CR_1	$w_c^1, w_{c+1}^1, w_{c+2}^1$	Combine $(c, c+1, c+2)$
CR_2	$w_c^1, w_{c+1}^1, w_{c+2}^{p\neq 1}$	Combine $(c, c+1)$
CR_3	$w_c^1, w_{c+1}^2, w_{c+2}^p$	Combine $(c, c+1)$
CR_4	$w_c^2, w_{c+1}^1, w_{c+2}^p$	Defined as in Figure 1.

3.3.2 Identification of English Transliteration Names

As stated in Chen and Lee [10], the identification of English transliteration names poses two challenges: Unlike Chinese personal names, there are no common surnames for identification. Common Chinese personal names are generally of length 2, 3, or 4. However, there is no restriction on the length of an English transliteration name. Examples such as "Walter Bruce Willis" (No. 2), "Kim Basinger" (No. 3) and "Steven Seagal" (No. 4) are shown in Table 6.

Although an English name may be translated into different Chinese names by different translators, Chen and Lee [10] believed that most translators use a popular character set for translation. Therefore, in our method, we developed an English Transliteration Character Set (ETCS) which contains 269 widely-used characters and is extracted from the common English names listed in Hi-Tutor [16]. In addition, our method greedily looks up the maximum of k such that all $\{w_c^p, w_{c+1}^p, \cdots, w_{c+k}^p\}$, where $k \geq 1$ and $c \leq c+k \leq m$, of k_j are in ETCS, then we combine $\{w_c^p, w_{c+1}^p, \cdots, w_{c+k}^p\}$ as one English transliteration name. Noted that, the tag variable p cannot be '*' and can be greater than 1.

3.3.3 Non-specific Type of Unknown Words

Due to non-specific types of unknown words without obvious clues to design rules, we design a variant (namely VBMA) of the bottom up merging algorithm, proposed by Chen and Ma [9], to extract more unknown words. The algorithm of VBMA is presented in Figure 2.

In Line 2 to Line 7 in Figure 2, a priority value for each morpheme w_i^p of k_j is calculated, and the priority sum of w_i^p and w_{i+1}^p is stored into Priority$_j[i]$. Then, the maximum value of Priority$_j[i]$, for all $0 \leq i < k_j$.length-1, is assigned to the variable *geti* of the token pair of morphemes of k_j with the highest priority value are denoted as w_{geti}^p and w_{geti}^p. Note that if there are more than one

```
Input  k_j to VBMA;
01 String VBMA(k_j)
02 constant String WordSpilt = ":"
03 For(int i = 0; i< k_j.length-1;i++){
04    Calculate the priority of  w_i^p  and  w_{i+1}^p
05    Save the priority into Priority_j[i]
06
07 geti = get the index value of the max(Priority_j[i]);
08 If Priority_j.length == 1
09   If (w_geti^p, w_{geti+1}^p) conforms to the general rules
10      return Combine(w_geti^p  +  w_{geti+1}^p);
11      else
12         return  w_geti^p  + wordSpilt +w_{geti+1}^p
13   else
14      If (w_geti^p, w_{geti+1}^p) conforms to the general rules
15         w_geti^ts  = Combine (w_geti^p, w_{geti+1}^p);
16         k_j  = {w_0^p ... w_geti^ts  w_{geti+2}^p ... w_m^p}
17         return VBMA(k_j);///recursive condition
18      else
19         if geti == 0
20            k_x  = {w_{geti+1}^p ... w_m^p};
21            return  w_geti^p + wordSpilt + VBMA(k_x);
22         else if geti == Priority_j Length-1
23            k_x  = {w_0^p ... w_geti^p};
24            return VBMA(k_x) + wordSpilt +w_{geti+1}^p  ;
25         else
26            k_{j-left}  = {w_0^p ... w_geti^p};
27            k_{j-right}  = {w_{geti+1}^p ... w_m^p};
28            return VBMA(k_{j-left}) + wordSpilt + VBMA(k_{j-right});
29       Output String  uk_j  = VBMA(k_j)
```

Figure 2: A variant of bottom up merging algorithm

pair with the identical max(Priority$_j[i]$), only the smallest *geti* is selected. In the case that the length of Priority$_j[i]$ is greater than 1, and (w_{geti}^p, w_{geti}^p) satisfies the general rules, w_{geti}^p with w_{geti+1}^p are combined as w_{geti}^{ts} (*ts* stands for token string), k_j becomes $\{w_0^p \cdots w_{geti}^{ts} w_{geti+2}^p \cdots w_m^p\}$, and execute VBMA($k_j$) recursively as shown in Line 15 to Line 17. If, on the other hand, (w_{geti}^p, w_{geti}^p) does not satisfy the general rules, k_j is split according to the value of geti as shown in Line 19 to Line 28. If *geti* is equal to 0, k_j is split into w_{geti}^p and $k_x = \{w_{geti+1}^p \cdots w_m^p\}$ as shown in Line 21. If *geti* is equal to Priority$_j$.length-1, k_j is split into $k_x = \{w_0^p \cdots w_{geti}^p\}$ and w_{geti+1}^p as shown in Line 24. If *geti* is neither 0 nor Prioirty$_j$.length-1, k_j is split into k_{j-left} and $k_{j-right}$ as shown in Line 26 and Line 27, respectively. The recursive method is performed as described above until Priority$_j$.length is equals to 1. As shown in Line 8 to Line 12, if Priority$_j$.length is equal to 1, and (w_{geti}^p, w_{geti}^p) satisfies the general rules, the combined (w_{geti}^p, w_{geti}^p) is returned. Otherwise, a string consisting of w_{geti}^p, ":", and w_{geti}^p is returned. After VBMA(k_j) is completed, it returns a string uk_j containing unknown words $jk_{j1} : jk_{j2} : \cdots : jk_{jz}$ separated by word delimiter ":"; that is, $uk_j = \{uk_{j1}, uk_{j2}, uk_{jy}, \cdots, uk_{jz}\}$ after removing the delimiter ":". Finally, every uk_{jy} is compared

to all w_c^p in k_j. If uk_{jy} is not identical to any w_c^p, uk_{jy} is identified as an unknown word.

1) Priority values
Chen and Ma [9] use a variant of mutual information, which is based on co-occurrence and mutual information, to calculate the priority value (i.e. probability) of any two consecutive morphemes. If there is more than one pair of two consecutive morphemes of the same priority, the first pair, reading w_c^p from left to right, is selected and combined. The co-occurrence, and mutual information, and the variant of mutual information are briefly described below:

a. Co-occurrence
The calculation of the co-occurrence of w_c^p and w_{c+1}^p is simply the number of times that w_c^p and w_{c+1}^p co-occur in QK_i and denoted as $f(w_c^p, w_{c+1}^p)$ listed below:

$$Co-occurrence(w_c^p, w_{c+1}^p) = f(w_c^p, w_{c+1}^p). \quad (1)$$

b. Mutual information
The calculation of mutual information of w_c^p, w_{c+1}^p, denoted as $MI(w_c^p, w_{c+1}^p)$, is shown in Equation (2) which was proposed by Gao and Lin [21]:

$$\begin{aligned} MI(w_c^p, w_{c+1}^p) &= \log_2 \frac{\frac{f(w_c^p, w_{c+1}^p)}{N}}{\frac{f(w_c^p)}{N} \times \frac{f(w_{c+1}^p)}{N}} \\ &\approx \log_2 \frac{N \times f(w_c^p, w_{c+1}^p)}{f(w_c^p) f(w_{c+1}^p)} \end{aligned} \quad (2)$$

where $f(w_c^p)$ is the frequency of w_c^p in QK_i, $f(w_{c+1}^p)$ and N is the total number of morphemes in QK_i. Moreover, N is 106 as suggested by Wu *et al.* [37].

c. A variant of mutual information
Chen an Ma [9] proposed a variant of mutual information (denoted as VMI) where VMI is equal to Co-occurrence (w_c^p, w_{c+1}^p) multiplying MI, and the equation is shown as follows:

$$VMI(w_c^p, w_{c+1}^p) = f(w_c^p, w_{c+1}^p) \log_2 \frac{N \times f(w_c^p, w_{c+1}^p)}{f(w_c^p) f(w_{c+1}^p)}$$

2) Unknown words extraction rules
The general rules for unknown word extraction are based on two kinds of conditions, namely pattern conditions and statistical conditions.

a. Pattern Conditions
The pattern conditions are summarized in Table 9. PC_1 represents a pattern such that the length of both w_c^p and w_{c+1}^p is 1; that is, (w_c^1, w_{c+1}^1). In PC_2, $w_{c+1}^{p=2,3}$ means the length of w_{c+1}^1 can be either 2 or 3. Take No. 1 in Table 8 as an example. The queried keyword No. 1 in Table 8 (i.e. General manager) is spilt into w_c^1 (i.e. General) and w_{c+1}^2 (i.e. manager). In PC_3, the ts of w_{c+1}^{ts} stands for token string which was defined previously. An example of PC_4 is shown in No. 2 of Table 8. The queried keyword No. 2 in Table 8 (i.e. Police Department) is spilt into w_c^2 (i.e. Police) and w_{c+1}^1 (i.e. Department). PC_{10} represents a pattern that an unknown word contains both English and Chinese; for examples, No. 2 and No. 3 in Table 3. However, PC_{10} only considers w_c^* and w_{c+1}^1, respectively. A pattern such as w_c^p and w_{c+1}^* is not considered because there is little, if none, unknown word is of that pattern. In addition, if p or ts is greater than 4, the morpheme is automatically treated as a known word.

b. Statistical Condition
Unlike other studies, the proposed method uses the frequency and probability of k_j in QK_i to design statistical rules; instead of the frequency and probability of a keyword in an article. Church [21] indicated that unknown words are keywords that are not in the LEX and often repeatedly appears in an article. Thus, SC_1, as shown in Table 10, is designed so that two consecutive morphemes are considered to be an unknown word if their frequency is greater than or equal to T_1, and T_1 is a threshold value. Also, Chen and Ma [9] pointed out that if w_c^p is a polysyllabic morpheme (i.e., $p > 1$ or $p = ts$), it is unlikely that two different unknown words contain the same w_c^p in one article. As a result, SC_2 and SC_3 are designed to take advantages of such feature. In SC_2, if the probability of w_c^p followed by w_{c+1}^p is greater than or equal to a threshold value T_2, w_c^p should be combined with w_{c+1}^p. Take No. 2 in Table 8 as an example. Based on SC_2, if the conditional probability of ("Department"—"Police") is greater than or equal to T_2, they will be combined as "Police Department". Similar to SC_2, if the probability of w_{c+1}^p followed by w_c^p is greater than or equal to T_2, w_c^p should also be combined with w_{c+1}^p. Take No. 1 in Table 8 as an example. Based on SC_3, if the conditional probability of ("General"—"manager") is greater than or equal to T_2, they will be combined as "General manager".

SC_4 and SC_5 indicate that a monosyllabic morpheme (either w_c^1 or w_{c+1}^1) is not in the Stop Word List (SWL). Based on the frequency, we selected the top 100 words from the Word List with Accumulated Word Frequency in Sinica Corpus 3.0 [36] as our SWL. However, there are 16 polysyllabic morphemes among them. Because polysyllabic morphemes can be unknown

words in the proposed scheme, they are removed from the SWL. Therefore, the number of morphemes of SWL is 84.

Table 7: Statistical conditions for general rules

Statistical Conditions ID	Statistical Conditions
SC_1	$f_{QK_i})(w_c^p w_{c+1}^p) \geq T_1$
SC_2	$\text{Probabilty}_{QK_i})(w_{c+1}^p \mid w_c^p) \geq T_2$
SC_3	$\text{Probabilty}_{QK_i}(w_c^p \mid w_{c+1}^p) \geq T_2$
SC_4	$w_c^1 \notin SWL$
SC_5	$w_{c+1}^1 \notin SWL$

Table 8: General rules

General Rules	Conditional Constraint
GR_1	$PC_1\&SC_4$
GR_2	$PC_2\&SC_1\&SC_3\&SC_4$
GR_3	$PC_3\&SC_3\&SC_4$
GR_4	$PC_4\&SC_1\&SC_2\&SC_5$
GR_5	$PC_5\&SC_1\&SC_2\&SC_3$
GR_6	$PC_6\&SC_1\&SC_2\&SC_3$
GR_7	$PC_7\&SC_1\&SC_2\&SC_3$
GR_8	$PC_8\&SC_2\&SC_5$
GR_9	$PC_9\&SC_2\&SC_3$
GR_{10}	$PC_{10}\&SC_1\&SC_2\&SC_5$

As shown in Table 8, we designed ten general rules according to the pattern conditions and statistical condition. If two consecutive morphemes satisfy any one conditional constrain, they are merged. For example, GR_1 show that, if two consecutive morphemes (w_c^p and w_{c+1}^p) are all monosyllabic morphemes (PC_1) and w_c^1 is not in the SWL (SC_4), w_c^1 and w_{c+1}^1 are combined to become w_c^{ts}. GR_5, GR_6, and GR_7 deal with two consecutive morphemes that are all polysyllabic morphemes. If two consecutive polysyllabic morphemes co-occur frequently (i.e., SC_1) and satisfy both SC_2 and SC_3, they should be combined to become w_c^{ts} as an unknown word. Take No. 5 in Table 8 as an example. Because w_c^{ts} is "Echo" and w_{c+1}^p is "Mr.", if the frequency of w_c^{ts} and w_{c+1}^p are greater than or equal to T_1, they are merged. GR_{10} is designed for unknown words that contain English and Chinese. Because w_c^* in PC_{10} is an English word such as "js" or "jar", w_{c+1}^1 should not be in the SWL (SC_5). In addition, if the frequency of w_c^* and w_{c+1}^1 is greater than or equal to T_1 (SC_1), and if the probability of w_c^* followed by w_{c+1}^p is greater than or equal to T_2 (SC_2), w_c^* and w_{c+1}^1 are combined to become w_c^{ts}.

4 Experimental Results

In the experiments, mmseg4j [35] is used as the Chinese segmentation system, in which its lexicon contains 124,499 entries. In addition, the proposed unknown word extraction method is used to improve mmseg4j, and it is called "improved mmseg4j" hereinafter. A total of 998 BC blog articles are retrieved from the BC database during 10/07/2009 and 07/31/2014. Among them, 112 articles do not have Chinese queried keywords; therefore, a total of 786 articles are used for the Chinese unknown word extraction in the experiments. For the 786 articles, the maximum, minimum, and average numbers of k_j in QK_i are 420, 1, and 7; respectively. The performance is evaluated by precision and recall, and they are defined as follows:

NC_i = The number of correctly extracted unknown words in QK_i;

NE_i = The number of extracted unknown words in QK_i;

NT_i = The total number of unknown words in QK_i;

m = The total number of keywords in QK_i;

Precision rate = $\frac{\sum_{i=1}^{m} NC_i}{\sum_{i=1}^{m} NE_i}$

Recall rate = $\frac{\sum_{i=1}^{m} NC_i}{\sum_{i=1}^{m} NT_i}$

F-measure = $2PR/(P+R)$.

The experiments are divided into two parts. The first part is to obtain the most appropriate threshold values (T_1 and T_2) for the improved mmseg4j. In the second part, we compare the performance of the improved mmseg4j with those of the original mmseg4j [35] and the CKIP [7].

4.1 The Experiments for Threshold Values

Co-occurrence, mutual information, and the variant of mutual information are used to calculate priority values. T_1 is set to either 2, 3, or 4; and T_2 is set to either 0.8, 0.9, or 1. The experimental results are summarized in Table 9. The name of the first column is "Priority-$T_1 - T_2$" which "Priority" can be either Co (i.e. co-occurrence), MI (i.e. mutual information), or VMI (i.e. the variant of mutual information); and T_1 and T_2 are threshold values. For example, Co_2_1 represents that the priority value is calculated based on co-occurrence, $T_1 = 2$, and $T_1 = 1$. As shown in Table 9, it is clear that the best F-measure values can be obtained when $T_1 = 2$ and $T_2 = 0.9$, regardless co-occurrence, mutual information, or the variant of mutual information are used. Additionally, when comparing the performance of Co_2_0.9, MI_2_0.9 and VMI_2_0.9, one can see that co-occurrence is better than mutual information and the variant of mutual information. This result is conformance to the experimental results presented in Chen and Ma [16]. Therefore, in

Table 9: The experimental results for T_1 and T_2

Evaluation / Prority-T_1-T_2	Precision	Recall	F-Measure
CO_2_1	84.81%	68.42%	75.74%
CO_2_0.9	85.18%	69.84%	76.75%
CO_2_0.8	82.67%	70%	75.8%
CO_3_1	84.9%	64.87%	73.54%
CO_3_0.9	85.35%	66.09%	74.49%
CO_3_0.8	82.80%	65.58%	73.19%
CO_4_1	85.63%	62.14%	72.02%
CO_4_0.9	85.46%	63.66%	72.96%
CO_4_0.8	83.6%	63.46%	72.15%
MI_2_1	81.04%	67.51%	73.65%
MI_2_0.9	81.76%	69.03%	74.85%
MI_2_0.8	80.51%	69.03%	74.32%
MI_3_1	82.47%	64.27%	72.24%
MI_3_0.9	82.82%	65.38%	73.07%
MI_3_0.8	81.62%	65.59%	72.27%
MI_4_1	82.74%	61.63%	70.06%
MI_4_0.9	83.22%	63.25%	71.87%
MI_4_0.8	82.09%	63.56%	71.16%
VMI_2-1	84.46%	68.22%	75.48%
VMI_2_0.9	84.95%	69.73%	76.59%
VMI_2_0.8	82.67%	70.04%	75.83%
VMI_3_1	84.22%	64.27%	72.9%
VMI_3_0.9	84.72%	65.68%	73.99%
VMI_3_0.8	82.68%	65.68%	73.21%
VMI_4_1	85.36%	61.94%	71.78%
VMI_4_0.9	85.21%	63.56%	72.80%
VMI_4_0.8	83.82%	63.46%	72.23%

the following experiments, the priority value is calculated based on the co-occurrence, and T_1 and T_2 are set as 2 and 0.9; respectively.

4.2 The Experiments For Performance Comparisons

In this paper, the unknown word extraction is significantly different from the past research because unknown words are extracted from the queried keywords that are incomplete sentences. Also, the size of the lexicon used by a Chinese segmentation system varies, and this in turn affects the performance of unknown word extraction. As a result, we calculate the number of unidentified unknown words for both the original mmseg4j and the improved mmseg4j.

CKIP, developed by Academia Sinica, is one of the most popular Chinese segmentation systems in Taiwan, and also won the championship of a competition held by ACL SIGHAN [7]. The lexicon of CKIP contains about 90,000 entries. In Table 10, we also calculate the number of unidentified unknown words when CKIP is used as the Chinese segmentation system.

As shown in Table 10, the improved mmseg4j is more effective than the original mmseg4j because 988 unknown words cannot be identified by the original mmseg4j, while 299 unknown words cannot be identified by the improved mmseg4j. Also, the number of unknown words that cannot be identified by CKIP is 415, which are 116 more than the improved mmseg4j. After further checking, we found that 278 unidentified unknown words of the 299 unidentified unknown words are not in the unidentified unknown words by the CKIP; and the remaining 21 unidentified unknown words are in the CKIP lexicon, but not mmseg4j. In other words, if the same lexicon is used, the number of unidentified unknown words can be further reduced to 278.

Table 10: The number of unidentified unknown words

mmseg4j	Improved mmseg4j	CKIP
988	299	415

5 Conclusions

This research is attempted to find an effective approach to extract Chinese unknown words from incomplete sentences. It would be very useful in applications such as search engines and social networks. The proposed scheme resolves the problems in two phases, namely unknown word detection phase and unknown word extraction phase. In the unknown words detection phase, we design rules to detect possible unknown words from, and save them into the unknown word candidate set. In the unknown words extraction phase, we designed both pat-

tern and statistical conditions to calculate priority values and proposed a variant of the bottom up merging algorithm to extract unknown words from. In the experiments, the precision, recall and F-measure up to 85.18%, 69.84%, and 76.75% can be obtained; respectively. Comparing to CKIP, one of the most popular Chinese segmentation system, the proposed scheme can extract 137 more unknown words. Based on our observation, we believe that the proposed scheme can be further extended using machine learning.

Acknowledgments

This work was supported by the Ministry of Science and Technology in Taiwan under Grant No. MOST 107-2221-E-182-081-MY3, and in part by the Kaohsiung Chang Gung Memorial Hospital.

References

[1] Academia Sinica Balanced Corpus, June 26, 2022. (http://db1x.sinica.edu.tw/cgi-bin/kiwi/mkiwi/kiwi.sh)

[2] U. Bojars, J. G. Breslin, V. Peristeras, G. Tummarello, S. Decker, "Interlinking the social Web with semantics," *Journal of IEEE Intelligent Systems*, vol. 23, no. 3, pp. 29-40, 2008.

[3] Blog Connect, June 26, 2022. (http://bridge.nchu.edu.tw/BC/)

[4] Y. H. Chen, E. J. L. Lu, M. F. Tsai, "Finding keywords in blogs: efficient keyword extraction in blog mining via user behaviors," *Expert Systems with Applications*, vol. 41, no. 2, pp. 663-670, 2013.

[5] Y. H. Chen, E. J. L. Lu, J. J. Huang, "Analysis Chinese segmentation systems on queried keywords," in *Proceedings of International Conference on Information Management (ICIM'14)*, 2014.

[6] K. J. Chen, S. H. Liu, "Word identification for Mandarin Chinese sentences," in *Proceedings of the 14th Conference on Computational Linguistics*, pp. 101-107, 1992.

[7] K. J. Chen, W. Y. Ma, "Introduction to CKIP Chinese word segmentation system for the first International Chinese word segmentation bakeoff," in *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing (SIGHAN'03)*, pp. 168-171, 2003.

[8] K. J. Chen, W. Y. Ma, "Unknown word extraction for Chinese documents," in *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02)*, pp. 1-7, Stroudsburg, PA, USA, 2002.

[9] K. J. Chen, W. Y. Ma, "A bottom-up merging algorithm for Chinese unknown word extraction," in *Proceedings of the second SIGHAN Workshop on Chinese Language Processing (SIGHAN'03)*, pp. 31-38, Stroudsburg, PA, USA, 2003.

[10] H. H. Chen, J. C. Lee, "Identification and classification of proper names in Chinese texts," in *Proceedings of the 16th conference on Computational linguistics*, pp. 222-229, 1996.

[11] K. J. Chen, C. J. Chen, "Knowledge extraction for identification of Chinese organization names," in *Proceedings of the Second Workshop on Chinese Language Processing: Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics*, pp. 15-21, Stroudsburg, PA, USA, 2000.

[12] K. J. Chen, M. H. Bai, "Unknown word detection for Chinese by a corpus-based learning method," *International Journal of Computational linguistics and Chinese Language Processing*, vol. 3, no. 4, pp. 27-44, 1998.

[13] T. H. Chang, C. H. Lee, "Automatic Chinese unknown word extraction using small-corpus-based method," in *Natural Language Processing and Knowledge Engineering*, pp. 459 - 464, Beijing, China, 2003.

[14] K. Church, W. Gale, P. Hanks, D. Hindle, "Using statistics in lexical analysis," in *Lawrence Erlbaum Associates Publishers*, pp. 115-164, 1991.

[15] CNNIC, June 26, 2022. (http://www.chinainternetwatch.com/whitepaper/china-internet-statistics/#ixzz3nUGYQuca)

[16] Common English names, June 26, 2022. (http://www.hitutor.com.tw/english-name.php)

[17] C. K. Fan, W. H. Tsai, "Automatic word identification in Chinese sentences by the relaxation technique," *Computer Proceeding of Chinese and Oriental Languages*, vol. 4, no. 1, pp. 33-56, 1988.

[18] S. Foo, H. Li, "Chinese word segmentation and its effect on information retrieval," *Information Processing & Management*, vol. 40, no. 1, pp. 161-190, 2004.

[19] J. Gao, W. Lai, "Formal concept analysis based clustering for blog network visualization," *Advanced Data Mining and Applications, Lecture Notes in Computer Science*, vol. 6440, pp. 394-404, 2010.

[20] C. L. Goh, M. Asahara, Y. Matsumoto, "Machine learning-based methods to Chinese unknown word detection and POS tag guessing," *Journal of Chinese Language and Computing*, vol. 16, no. 4, pp.185-206, 2006.

[21] J. M. Gao, C. L. Lin, Corpus Constriction, June 26, 2022. (http://www.naer.edu.tw/ezfiles/0/1000/img/25/439422325.pdf)

[22] X. Jiang, L. Wang, Y. Cao, Z. Lu, "Automatic recognition of Chinese unknown word for single-character and affix models," in *Proceedings of the Sixth International Conference on Intelligent Systems and Knowledge Engineering*, pp. 435-444, Shanghai, China, 2011.

[23] N. Johnson, *Google on User Intent in Search Queries, Search Engine Watch*, June 26, 2022. (http://searchenginewatch.com/article/2053806/Google-On-User-Intent-in-Search-Queries)

[24] K. Lei, W. Y. Zhang, K. Zhang, K. Xu, "Extracting Unknown Words from Sina Weibo via Data Clustering," in *IEEE International Conference on Communications (ICC'15)*, pp. 1182-1187, 2015.

[25] B. Larsen, C. Aone, "Fast and effective text mining using linear-time document clustering," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 16-22, New York, NY, USA, 1999.

[26] H. Li, C. H. Huang, J. Gao, X. Fan, "The use of SVM for Chinese new word identification," *Lecture Notes in Computer Science*, vol. 3248, pp 723-732, 2005.

[27] H. Li, B. Yuan, "Chinese word segmentation," in *Proceedings of the 12th Pacific Asia Conference on Language Information and Computation*, Singapore, pp. 212-217, 1998.

[28] Y. Lai, C. Wu, "Unknown word and phrase extraction using a phrase-like-unit based likelihood ration," *International Journal of Computer Processing*, vol. 13, no. 1, pp. 83-95, 2000.

[29] B. I. Li, "A maximal matching automatic Chinese word segmentation algorithm using corpus tagging for ambiguity resolution," in *Computational Linguistics Conference*, pp. 135-146, 1991.

[30] J. Nie, M. Briscbois, X. Ren, "On Chinese text retrieval," in *Proceedings of the 19th annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 225-233, New York, NY, USA, 1996.

[31] L. Qiu, Y. Zhang, "Word Segmentation for Chinese Novels," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 2440-2446, 2015.

[32] Sina Weibo, June 26, 2022. (`http://www.weibo.com/login.php`)

[33] J. Sobel, *State of the Blogosphere 2011: Introduction and Methodology*, June 26, 2022. (`http://www.199it.com/archives/tag/state-of-the-blogosphere-2011`)

[34] Y. T. Sung, T. H. Chang, W. C. Lin, K. S. Hsieh, K. E. Chang, "CRIE: An automated analyzer for Chinese texts," in *Behavior Research Methods*, pp. 1-14, 2015.

[35] C. H. Tsai, *MMSEG4J: A Word Identification System for Mandarin Chinese Text Based on Two Variants of the Maximum Matching*, June 26, 2022. (`http://technology.chtsai.org/mmseg4j/`)

[36] Word List with Accumulated Word Frequency in Sinica Corpus 3.0, June 26, 2022. (`http://www.aclclp.org.tw/doc/wlawf_abstract.pdf`)

[37] Y. Wu, C. Hsieh, W. Lin, C. Liu, L. Yu, "Unknown word extraction from multilingual code-switching sentences," in *Proceedings of the 23rd Conference on Computational Linguistics and Speech Processing*, pp. 349–360, Stroudsburg, PA, USA, 2011.

[38] C. C. Yang, C. H. Chang, "A two-phase approach to Chinese unknown word extraction: Application of pattern mining and machine learning," in *The 13th conference on Artificial Intelligence and Application*, Tamkang University Lanyang Campus, 2008.

[39] Y. Ye, Q. Y. Wu, Y. Li, K. P. Chow, L. C. K. Hui, S. M. Yiu, "Unknown Chinese word extraction based on variety of overlapping strings," *Information Processing & Management*, vol. 49, no. 2, pp. 497-512, 2013.

Biography

Yi-Hui Chen received her Ph.D. degree in computer science and information engineering at the National Chung Cheng University. Next, she worked at Academia Sinica as a post-doctoral fellow. Later on, she worked at IBM's Taiwan collaboratory research center as a research scientist. She worked at the department of M-commerce and multimedia Applications, Asia University until 2019. She now works at department of information management, Chang Gung University. Her research interests include semantic web, text mining, and multimedia security.

Eric Jui-Lin Lu received the B.A. degree from the National Chiao-Tung University. Later on, He received his MSBA from San Francisco State University, U.S.A. He got his Ph.D. (Computer Science) at Missouri University of Science and Technology (formerly University of Missouri-Rolla), U.S.A. in 1996. He is currently a professor at the Department of Management Information Systems, National Chung Hsing University. His research interests include image and semantic web, social network, text mining, and networking.

Jeng Jie Huang received the M.S. degree in Department of Management Information Systems, National Chung Hsing University. His research interests include text processing and text mining.

A New Approach for Power Signal Disturbances Classification Using Deep Convolutional Neural Networks

Yeong-Chin Chen, Sunneng Sandino Berutu, Long-Chen Hung, and Mariana Syamsudin
(Corresponding author: Yeong-Chin Chen)

Department of Computer Science and Information Engineering, Asia University
Taichung 41354, Taiwan
Email: ycchenster@gmail.com

(Received Jan. 25, 2021; Revised and Accepted May 12, 2022; First Online June 29, 2022)

Abstract

This paper proposes a new approach for power signal disturbances (PSDs) classification using a two-dimension (2D) deep convolutional neural network (CNN). The data preprocessing stage introduces a conversion method from signal to the 2D grayscale image. Firstly, the signal is divided into multiple cycles. The zero-crossing rate is adopted to specify a cycle's start and endpoints. Then, the cycles are transformed into matrices. Next, the matrices are merged into a new form matrix. Lastly, the matrix is converted into the 2D image grayscale. The obtained 2D image preserves information and waveform the sinusoidal of the signal. The experiment was carried out on datasets containing 14 different disturbance categories with the same model learning structure. The results show that the 2D deep CNN performs better than the one-dimension (1D) deep CNN. According to this result, the 2D deep CNN can improve the PSDs classification effectiveness. Furthermore, the proposed method outperforms the conversion method used in previous studies.

Keywords: 2D Deep CNN; Conversion; Image; Signal Disturbance

1 Introduction

Power quality refers to interference-free electricity signals. Various deviations caused from loads [11,18] are to be considered as power signal disturbances (PSDs). The emergence of distortions in the power quality strongly affects the decreasing performance or malfunction of electricity equipment at industry, office, and home. In addition, the disturbances can cause an economic loss because of the reparation and replacement cost of the equipment damage. Therefore, identifying and classifying the PSDs are the best method in those worst impact avoiding. Scientific research has been carried out to address this problem. The rapid advancement of the deep learning method has been implemented in the PSDs field. Convolutional neural network (CNN) has the most performance capability and is widely employed in the PSDs classification work [18]. The one-dimensional (1D) CNN and the two-dimensional (2D) CNN methods have been implemented in the PSDs field. The 1D CNN is applied for the 1D dataset, whereas the 2D CNN is fed by the 2D dataset. Recently, the most popular CNN in the PSDs classification task is the deep CNN method. This method has higher performance in comparison with the others [21].

The 1D CNN in PSDs classification has been implemented by many authors [1, 5, 15, 17, 20, 21]. These works have proposed new approaches to improve the classification performance such as addressed the over fitting problem [21], improved feature extraction [1, 17], and introduced a hybrid model [15]. The performance shortages such computational time and model size are tried to be solved by implementing data compression techniques in the data preprocessing [5, 20]. However, these investigations consumed a lot of original information in the compression process.

In the beginning, the CNN method was employed for the 2D image classification purpose [16]. The 2D CNN method can learn the diversity and complexity of image features [12]. When the 2D CNN is implemented for the PSDs classification task, the 2D dataset is required for this method. However, the power signal data is one-dimensional and represented in sinusoidal waveforms. Therefore, a data preprocessing is required to convert from the power signal to the 2D image. Various conversion techniques were carried out by authors in references [2–4, 9, 10, 13, 14, 22, 24, 25]. The author in [13] employed a trajectory matrix to produce a lag-covariance model as image of PSDs. In addition, the work in [4] utilized quadratic means to generate the disturbance image. The other studies in [2, 3, 22] adopted a space phasor diagram (SPD) to transform the sag disturbance into the image. Besides, the investigations in [9, 10, 25] uti-

lized a matrix to transform the signal disturbance into the 2D image. The sampling points of signal are rearranged into a number rows and columns in the matrix, then convert the matrix into the gray-scale image. The works in [14, 24] adopted a scalogram and spectrogram analysis to represent the signal in the image. However, the transforming process has changed the original information totally [2, 3, 22], thus several important features are lost. The image size resulted in [14] is a large and the training time costly. In addition, the performance comparison of the 1D CNN and the 2D CNN models for the PSDs classification is unevaluated in the previous investigations.

In this study, a robust data preprocessing method is developed to convert from the signal to the 2D gray-scale image, where the image results can represent the sinusoidal waveform and preserve the original information. The 2D image obtained is used as the 2D dataset in the 2D deep CNN for the PSDs classification purpose. Moreover, the performance comparison of the 1D and 2D deep CNN models for PSDs classification is evaluated utilizing a confusion matrix method. In addition, to compare the efficacy of proposed conversion approach, the conversion methods [9, 10, 25] are implemented using same the 1D signal and same the 2D deep CNN architecture. The rest of this paper is organized as follows. First, Section 2 presents the material of this work and methods utilized for signal conversion and the PSDs classification. Section 3 shows the experimental result and discussion. Finally, conclusion and future study are explained in Section 4.

2 Material and Methods

In this section, first, the mathematical formula for generation of PSDs data is explained. Furthermore, the approach of conversion signal-to-image proposed is presented. Then, the deep CNN model structure is discussed.

2.1 Mathematical Formula of PSDs

With the limitation of the real PSDs data, this work employed the mathematical formulas [8, 20, 21] to generate the synthetic PSDs. In these equations, the IEEE-1159 standard parameter variations [7] are adopted. As presented in Table 1, this work utilizes 14 categories of disturbance signal.

The parameters value such as intensity (α), distortion of the transient (β), distortion of the flicker (λ), time (t_1 and t_2) are generated randomly to obtain the variety of each disturbance category. The fundamental frequency (f) is adjusted at 60 Hz, whereas the sampling frequency (fs) is 3200 Hz [17], the cycle numbers (Nc) is 11, the sampling points (Ns) is 586, and the amplitude (A) is set at 1. The synthetic signals produced for each category are 11,000 samples so that the total samples are 154,000.

2.2 The Signal to Image Conversion Approach

In this approach, the signal is divided into multiple cycles, where zero-crossing rate (ZCR) is utilized to determine the start and endpoints of cycles. The cycles are transformed into the matrices. The matrices are then merged to form a new matrix. The matrix result is converted to the 2D grayscale image. The advantage of this approach is that the image resolution can be reduced. The main steps of the proposed approach are depicted in Figure 1.

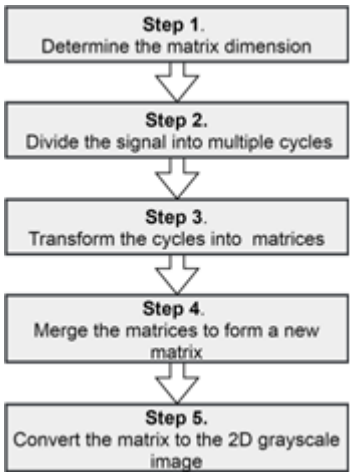


Figure 1: The steps of the conversion from the signal to the 2D grayscale image

The detailed explanation of Figure 1 is presented as follows:

Step 1. Determine the matrix dimension.

The square matrix (number of the rows (Nr) is equal to the number of the columns ($Ncol$) is chosen. The $Ncol$ is determined using Equation (1),

$$Ncol = ceiling(\frac{fs}{f}) \quad (1)$$

If fs and f values are 3200 and 60, respectively, so that $Ncol$ value is 54. Then, the matrix dimension is to be 54×54.

Step 2. Divide the signal into multiple cycles.

The signal is divided into 11 cycles, with the start and endpoints of each cycle determined by the ZCR. The rate at which the signal changes from negative to zero to positive is adopted in this work. As shown in Figure 2, the ZCR points obtained are marked in the signal.

According to Figure 2, the number of ZCR points obtained is 11, where the sampling points of the signal as ZCR are 1, 54, 107, 161, 214, 267, 320, 374, 427, 480, and 534. Therefore, the start and endpoints of each cycle can be obtained which presented in Table 2.

Table 1: Mathematical model and parameter of power signal disturbances

Categories	Mathematical formulas	Parameters
Normal	$y(t) = A[1 \pm \alpha(u(t-t_1) - u(t-t_2))]sin(\omega t)$	$\alpha \le 0.1$, $T \le t_2 - t_1 \le 9T$, $\omega = 2\pi f$
Sag	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))]sin(\omega t)$	$0.1 \le \alpha \le 0.9$, $T \le t_2 - t_1 \le 9T$
Swell	$y(t) = A[1 + \alpha(u(t-t_1) - u(t-t_2))]sin(\omega t)$	$0.1 \le \alpha \le 0.8$, $T \le t_2 - t_1 \le 9T$
Interruption	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))]sin(\omega t)$	$0.9 \le \alpha \le 1$, $T \le t_2 - t_1 \le 9T$
Harmonics	$y(t) = A[\alpha_1 sin(\omega t) + \alpha_3 sin(3\omega t) + \alpha_5 sin(5\omega t) + \alpha_7 sin(7\omega t)]$	$0.05 \le \alpha_3, \alpha_5, \alpha_7 \le 0.15$, $\sum \alpha_i^2 = 1$
Flicker	$y(t) = A[1 + \lambda sin(\omega_f t)]sin(\omega t)$	$8 \le f_f \le 25\ Hz$, $w_f = 2\pi f_f$ $0.05 \le \lambda \le 0.1$
Transient oscillation	$y(t) = A[sin(\omega t) + \beta e^{-(t-t_1)/\tau} sin(\omega_n(t-t_1))(u(t-t_2) - u(t-t_1))]$	$300 \le f_n \le 900$, $\omega_n = 2\pi f_n$,$0.5T \le t_2 - t_1 \le \frac{Nc}{3.33}T$, $8\ ms \le \tau \le 40\ ms$, $0.1 \le \beta \le 0.8$
Periodic notch	$y(t) = sin(\omega t) - sign(sin(\omega t)) \times \{\sum_{n=0}^{9} k\,[u(t - (t_1 - sn) - u(t - (t_2 - sn))]\}$	$0.01T \le t_2 - t_1 \le 0.05T$, $t_2 \le s$, $t_1 \ge 0$, $0.1 \le k \le 0.4$, c={1,2,4,6}, $s = \frac{T}{c}$
Sag with harmonics	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))][\alpha_1 sin(\omega t) + \alpha_3 sin(3\omega t) + \alpha_5 sin(5\omega t)]$	$0.1 \le \alpha \le 0.9$, $T \le t_2 - t_1 \le 9T$, $0.05 \le \alpha_3, \alpha_5, \alpha_7 \le 0.15$, $\sum \alpha_i^2 = 1$
Swell with harmonics	$y(t) = A[1 + \alpha(u(t-t_1) - u(t-t_2))][\alpha_1 sin(\omega t) + \alpha_3 sin(3\omega t) + \alpha_5 sin(5\omega t)]$	$0.1 \le \alpha \le 0.8$, $T \le t_2 - t_1 \le 9T$ $0.05 \le \alpha_3, \alpha_5, \alpha_7 \le 0.15$, $\sum \alpha_i^2 = 1$
Interruption with harmonics	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))][\alpha_1 sin(\omega t) + \alpha_3 sin(3\omega t) + \alpha_5 sin(5\omega t)]$	$0.9 \le \alpha \le 1$, $T \le t_2 - t_1 \le 9T$ $0.05 \le \alpha_3, \alpha_5, \alpha_7 \le 0.15$, $\sum \alpha_i^2 = 1$
Flicker with harmonics	$y(t) = A[1 + \lambda sin(\omega_f t)][\alpha_1 sin(\omega t) + \alpha_3 sin(3\omega t) + \alpha_5 sin(5\omega t)\,]$	$0.05 \le \lambda \le 0.1$, $8 \le f_f \le 25\ Hz$, $0.05 \le \alpha_3, \alpha_5, \alpha_7 \le 0.15$, $\sum \alpha_i^2 = 1$
Flicker with sag	$y(t) = A[1 + \lambda sin(\omega_f t)(1 - \alpha(u(t-t_1) - u(t-t_2))]sin(\omega t)$	$0.1 \le \alpha \le 0.9$, $T \le t_2 - t_1 \le 9T$, $0.05 \le \lambda \le 0.1$, $8 \le f_f \le 25\ Hz$
Flicker with swell	$y(t) = A[1 + \lambda sin(\omega_f t)(1 + \alpha(u(t-t_1) - u(t-t_2))]sin(\omega t)$	$0.1 \le \alpha \le 0.8$, $T \le t_2 - t_1 \le 9T$, $0.05 \le \lambda \le 0.1$, $8 \le f_f \le 25\ Hz$

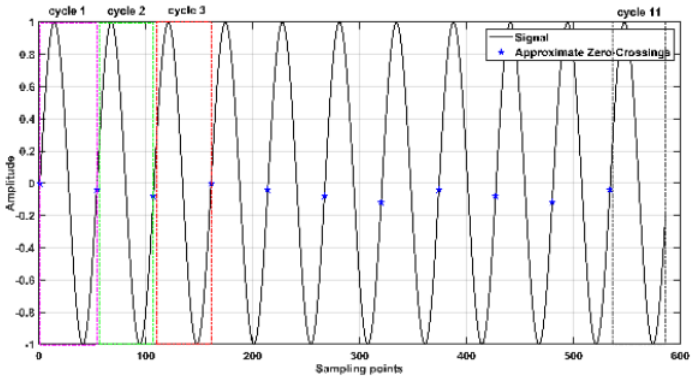


Figure 2: The zero-crossing rate points in the signal

Table 2: The start and endpoints of each cycle

Cycle	Start point	End point
1th	1	54
2nd	55	107
3rd	108	161
4th	162	214
5th	215	267
6th	268	320
7th	321	374
8th	375	427
9th	428	480
10th	481	534
11th	534	586

Step 3. Transform the cycles into matrices.

The cycle is transformed into a matrix of dimension 54×54. The start and the endpoints of the cycle are adopted as the columns of the matrix. In contrast, the sampling value of each point is used to determine the rows of the matrix. The sampling value of these points is then entered into the matrix elements. The following are the specifics:

1) Set the zero matrix:
Initially, the elements of matrix are set at 0.

2) Indicate the column numbers:
The start and endpoints of a cycle are indexed as column numbers to the matrix of dimension 54×54.

3) Arrange the sampling values into multiple classes:
The sampling values of the signal are arranged into different classes. The number of classes should be the same as the number of rows, and the width of the classes should be the same as well. The width of the class interval (Int) is calculated with Equation (2). In this case, the row number refers to the class number.

$$Int = \frac{Hs - Ls}{Nr} \tag{2}$$

In which Hs represents the highest sampling value, whereas Ls represents lowest sampling value from all the sampling values. Furthermore, the lower (LB) and upper (UB) boundaries are used to define the class interval limits. The boundaries of each class are obtained through steps which depicted in Figure 3. The

order of classes is started from the highest sampling value as the first class, while the lowest sampling value is in the 54rd class.

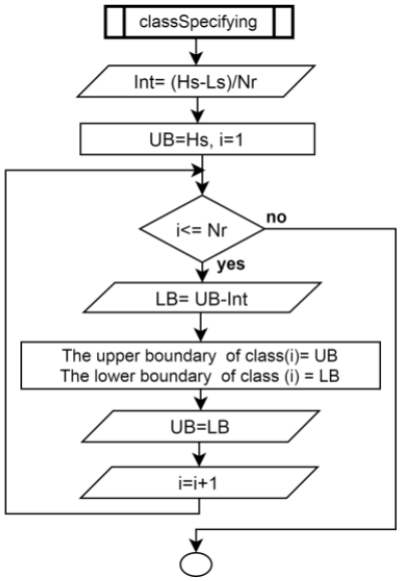


Figure 3: Steps of boundary determination for each classes

4) Specify the row numbers:
According to the classes resulted in Step 3.3, the row number of each sampling point can be obtained by comparing the sampling value to all the classes. The stages to determine the row number of each sampling point is presented in Figure 4.

5) Insert the sampling values of a cycle as the matrix elements:
The sampling values are inserted as the elements of matrix according to the row and column number which obtained at Steps 3.2 and 3.4.

Steps 3.1, 3.2, 3.4, and 3.5 are repeated to transform the rest cycles into the matrices.

Step 4. Merge the matrices to form a new matrix.
These matrices are combined by the add matrix function to form a new matrix with the same dimensions.

Step 5. Convert the matrix to the 2D grayscale image.
The elements of the matrix are converted to the grayscale color (0-255) to create the grayscale image. The image resolution result is 54×54 pixels.

2.3 Deep CNN Structure

The 1D and 2D of deep CNN methods were employed to classify the PSDs. The 1D convolution is utilized to classify the 1D signal, whereas the 2D convolution layer is implemented for the 2D image dataset. As depicted in Figure 5, the deep CNN structure is composed of 6 convolution layers, 3 max pooling layers, a dropout layer, and 2 dense of fully connected layers. The detail of these compositions is presented in Table 3.

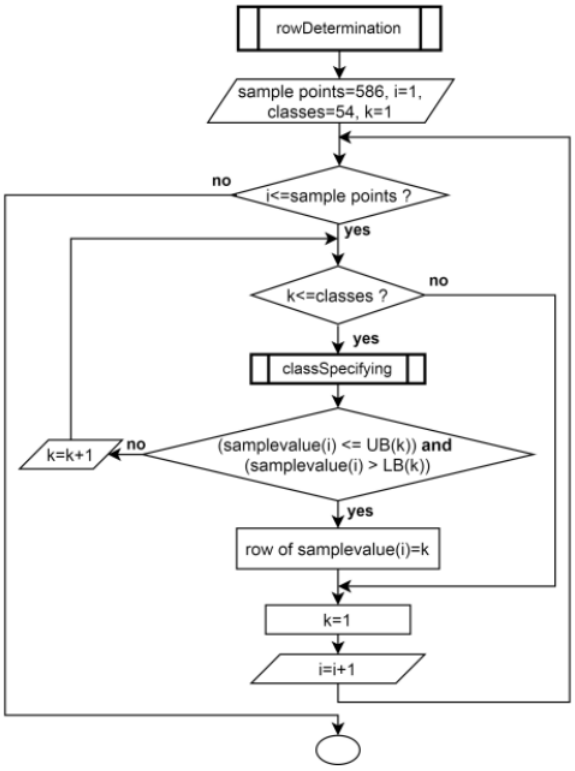


Figure 4: The steps of row specifying for each sampling point

2.4 Model Evaluation

The confusion matrix is employed to measure the parameters such as accuracy, recall, precision, and f1-score [6, 19, 23]. The four categories output of the confusion matrix such as true positive (TP), false positive (FP), true negative (TN), and false negative (FP) are calculated to obtain these parameters values. The parameters are used to evaluate the classification performance of the 1D and 2D deep CNN models.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$recall = \frac{TP}{TP + FN} \quad (4)$$

$$precision = \frac{TP}{TP + FP} \quad (5)$$

$$f1 - score = \frac{(2 \times precision \times recall)}{(precision + recall)} \quad (6)$$

3 Results and Discussion

In this section, first, the results of our approach for the signal to image conversion were presented. Then, the datasets used in this work are described. Furthermore,

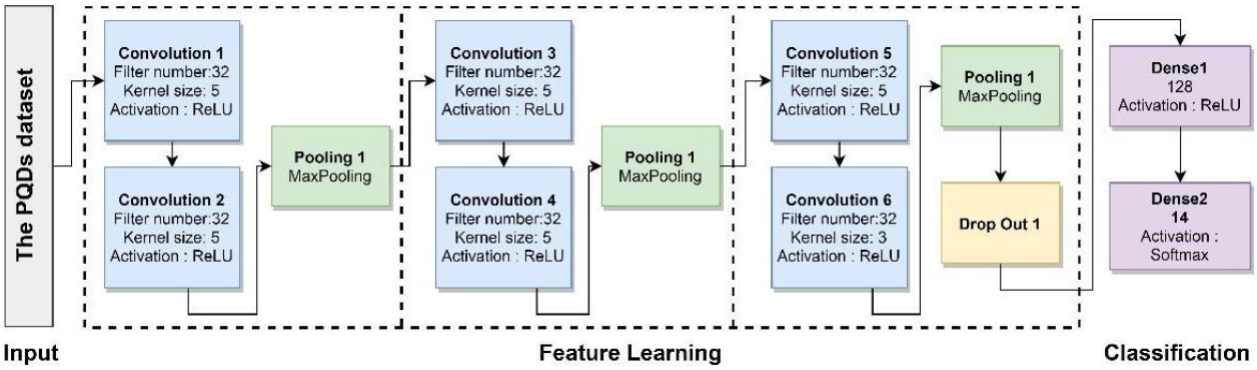


Figure 5: The architecture of the deep CNN model

the models training and testing stages are explained. The results of the training and testing are analyzed to evaluate the model's performance.

3.1 Implementation of the Signal to Image Conversion

The 14 synthetic disturbance types were generated using the mathematical model from Table 1. Then, the signal is converted to the 2D grayscale image utilizing our conversion proposed. As shown in Table 4, the 2D image obtained represents the sinusoidal waveform where the cycles of a signal are located in the image. In addition, the original amplitude values can be preserved in the image, although these values are converted into the grayscale color.

3.2 Datasets

In this work, the 1D signal dataset was obtained from the implementation result of the mathematical formula in Table 1, whereas the 2D image dataset was acquired from applying our approach for a conversion of the 1D signal to 2D image. In addition, we also employed the existing conversion methods [9,10,25] to obtain two 2D image datasets. Thus, three 2D image datasets are utilized in this work which presented in Table 5.

The 2D grayscale image sizes of the X, Y, and Z datasets are 54x54, 24 x 24, and 30 x 20, respectively. The 2D image dataset resulted from the previous methods are used to evaluate our approach performance. For the training and validation purpose, we used 9,900 samples per category, whereas about 500 samples of each type are utilized in the testing phase. The total samples of each dataset are 145,600. The details of dataset splitting for training, validation, and testing are presented in Table 6.

3.3 Training Stage Results

The model structure in Table 3 is utilized for the training phase. The 1D deep CNN model was trained using the 1D dataset, whereas the 2D deep CNN models were fed using the X, Y, and Z datasets. In the models, an Adam optimizer with a learning rate of 0.001 is adopted. Whereas, a categorical cross-entropy is employed for the loss function. In addition, the batch size is adjusted at 32. In the 2D deep CNN models, a rescaling layer is set at the first layer in the structure. Furthermore, a Nvidia Tesla T4 GPU accelerator 16 GB memory, and Intel Xeon (R) Central Processing Unit (CPU) @ 2.20 GHz are the training model environments.

In the beginning, the models were trained at 100 epochs. However, the accuracy and loss values of training and validation after the 50 epoch are shown unstable. Therefore, the models were retrained at 50 epochs. In addition, the dropout layer values of each model are adjusted to achieve the fitting accuracy and loss values between training and validation in the models. The dropout values for the 1D deep CNN, the 2D deep CNN X, the 2D deep CNN Y, and the 2D deep CNN Z are set at 0.55, 0.37, 0.45, and 0.55, respectively. Finally, the evaluation of the performance model training of the 1D and 2D deep CNN presents in Table 7. The fitting graph between the training and validation of models are displayed in Figure 6.

As presented in Table 7, generally, the performance of the 1D deep CNN outperforms both in the accuracy of training and validation than the 2D deep CNN models. In addition, the validation accuracy values are a higher than the training accuracy for all models. In the 2D deep CNN, the accuracy value of the 2D deep CNN X model exceeds the others. It indicates that the proposed approach performance in the conversion task is better than the previous approaches.

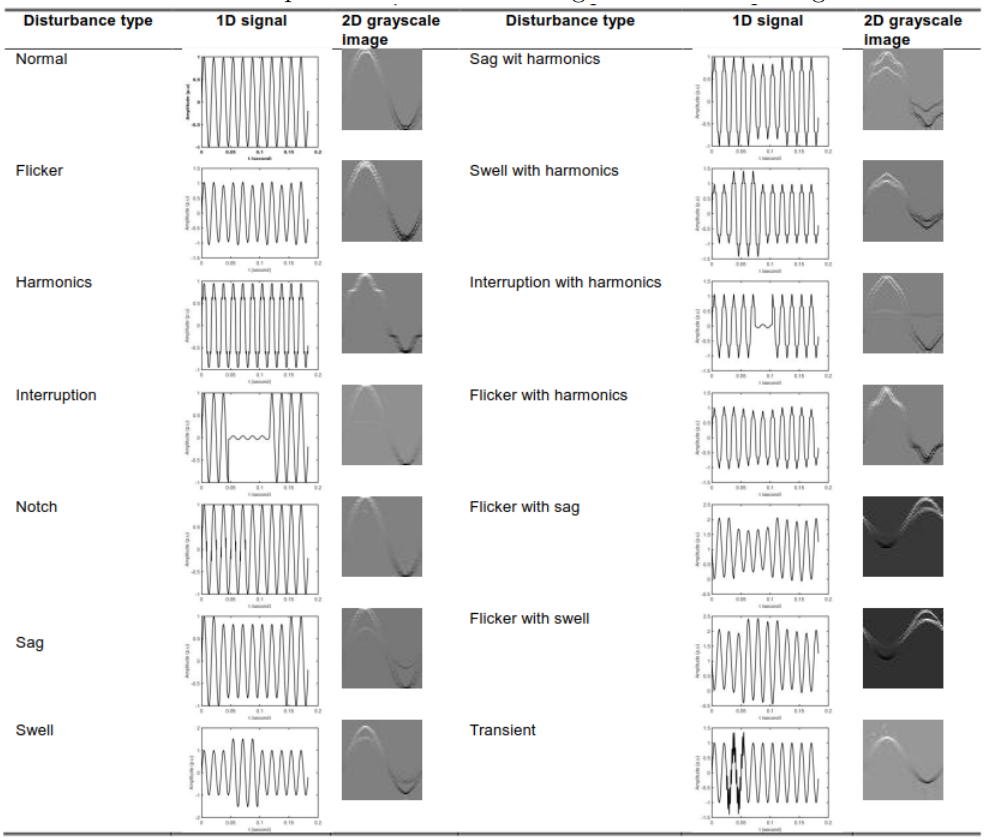
3.4 Discussion

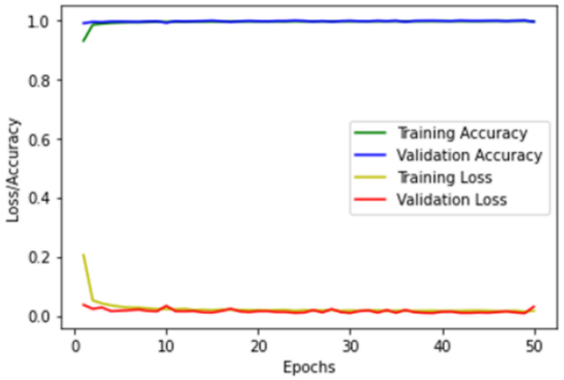
In the models evaluation stage, the 1D deep CNN was tested using 7,000 samples of the 1D signal, whereas the 2D deep CNN models were examined with 7,000 samples each which were obtained from our approach, the author's method in [9, 10], and in [25]. The results of each model testing are presented in the confusion matrices which are shown in Figure 7. From these confusion matrices, the parameters value such the recall, the precision, and the

Table 3: The detail of model architecture

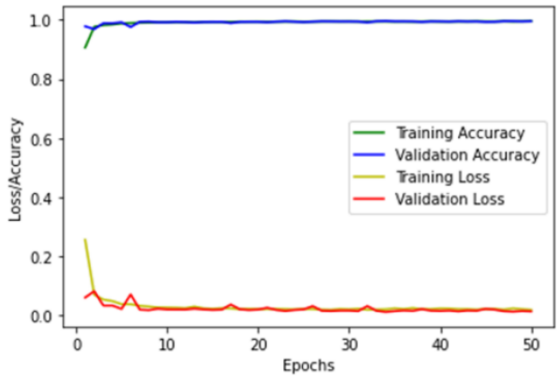
Layer	The 1D deep CNN	The 2D deep CNN
Convolution 1	Conv1D (32,5), activation = rectified linier unit (ReLU)	Conv2D (32,5) , activation=ReLU
Convolution 2	Conv1D (32,5) , activation=ReLU	Conv2D (32,5) , activation=ReLU
Pooling 1	Maxpooling1D(2)	Maxpooling2D(2)
Convolution 3	Conv1D (32,5) , activation=ReLU	Conv2D (32,5) , activation=ReLU
Convolution 4	Conv1D (32,5) , activation=ReLU	Conv2D (32,5) , activation=ReLU
Pooling 2	Maxpooling1D(2)	Maxpooling2D(2)
Convolution 5	Conv1D (32,5) , activation=ReLU	Conv2D (32,5) , activation=ReLU
Convolution 6	Conv1D (32,5) , activation=ReLU	Conv2D (32,5) , activation=ReLU
Pooling 3	Maxpooling1D(2)	Maxpooling2D(2)
Dense 1	Units = 128, activation=ReLU	Units = 128, activation=ReLU
Dense 2	Units = 14, activation= softmax	Units = 14, activation= softmax

Table 4: Representation of the 1D signal and the 2D image

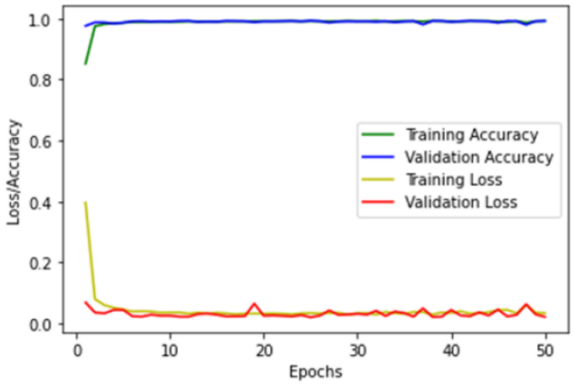




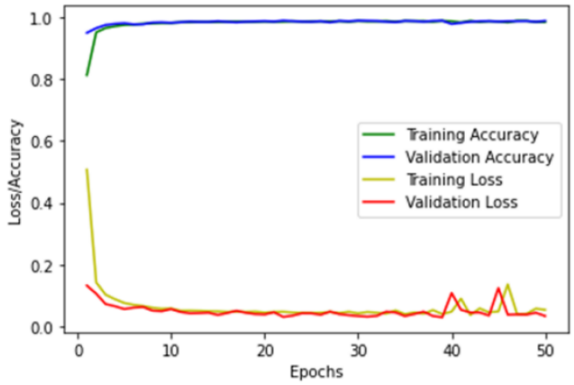
(a)



(b)



(c)



(d)

Figure 6: Fitting model of (a). the 1D deep CNN, (b). the 2D deep CNN X, (c). the 2D deep CNN Y, (d). the 2D deep CNN Z

Table 5: The 2D dataset and model name

Conversion method	Dataset	Model
Our proposed approach	X	2D deep CNN X
Author's approach [9, 10]	Y	2D deep CNN Y
Author's approach [25]	Z	2D deep CNN Z

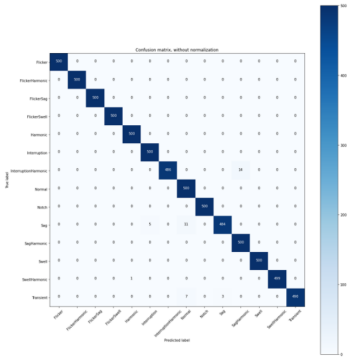
Table 6: Splitting of the 1D and 2D dataset

	1D signal	2D grayscale image
Training set	110,880	110,880
Validation set	27,720	27,720
Testing set	7,000	7,000

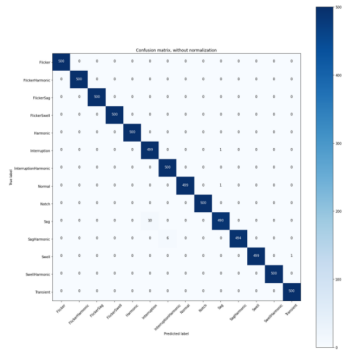
Table 7: Models performance in the training phase

Models Models	Training accuracy (%)	Validation accuracy (%)
1D deep CNN	99.27	99.51
2D deep CNN X	99.10	99.23
2D deep CNN Y	98.68	98.91
2D deep CNN Z	97.97	98.33

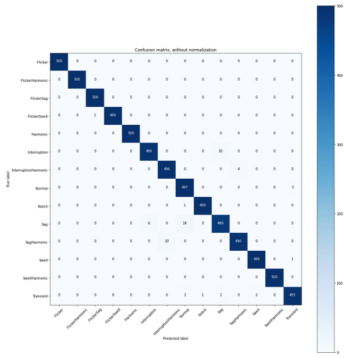
f1- score are obtained.



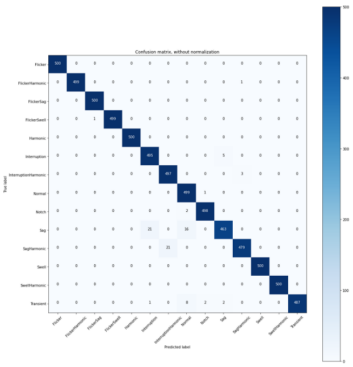
(a)



(b)



(c)



(d)

Figure 7: Confusion matrix of (a). the 1D deep CNN, (b). the 2D deep CNN X model, (c). the 2D deep CNN model, (d). the 2D deep CNN Z model

Firstly, we evaluated the testing performance of the 1D deep CNN and the 2D deep CNN X models. The parameters value of each disturbance are presented in Table 8 and Figure 8. The experiment's result showed that the flicker category and its combination achieved 100% for all the parameters value of both the models. These results are also obtained by the authors in [5, 21]. As the confusion matrices presented in Figure 7(a) and 8(b), the testing resulted of the 1D deep CNN, the number of disturbances in which the TP values reaching 100% are ten categories, whereas the 2D deep CNN X obtains nine categories. On the other hand, the lowest TP value of the 1D deep CNN is the interruption harmonic at 97.2%, where the rest (FN) is detected as the sag harmonic. Meanwhile, in the 2D deep CNN X model, the sag category is the lowest with 98%, where the rest (FN) is identified as the interruption disturbance. It can occur because the minimum boundary value of the intensity (α) interruption is equal to the maximum boundary of the sag disturbance.

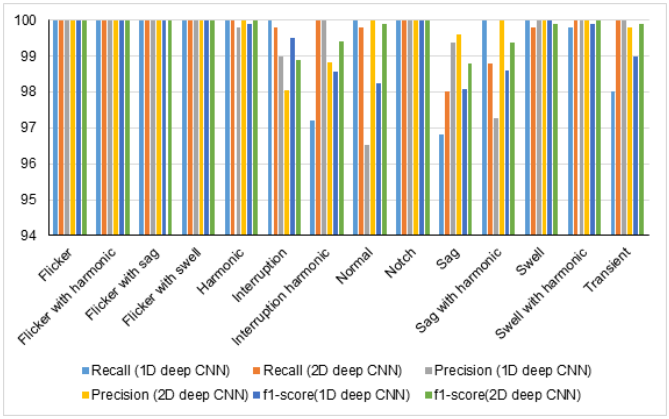


Figure 8: Bar chart of the testing evaluation between the 1D and 2D deep CNN X

As presented in Table 9 and Figure 9, generally, the value of the parameters of the 2D-X model exhibits better performance than the 1D deep CNN. The 2D deep CNN X model obtains 99.96% for the accuracy. The precision is acquired at 99.73%. The recall and f1-score reach 99.72% each. In addition, the size of the dataset and the model file are small. However, the 2D deep CNN X model takes relatively a cost computation time in training stage.

Furthermore, we verified the robustness of our approach in comparison to the previous approaches. An evaluation of the classification performance of the models using the 2D datasets from our approach and approaches used in the previous research is given in Table 10 and Figure 10. The experiment's result demonstrated that the 2D deep CNN Y model obtains 99.88% for accuracy, 99.19% for precision, 99.18% for recall, and 99.18% for f1-score. The 2D deep CNN Z reaches 99.74% for accuracy, 98.80% for precision, 97.81 for recall, and 98.25 for f1-score. It can be seen that our proposed approach outperforms other methods with 99.96% for accuracy, 99.73%

Table 8: Model performance of the 1D deep CNN (1D) and 2D deep CNN X (2D)

Disturbance categories	Recall (%)		Precision (%)		f1-score (%)	
	1D	2D	1D	2D	1D	2D
Flicker	100	100	100	100	100	100
Flicker with harmonic	100	100	100	100	100	100
Flicker with sag	100	100	100	100	100	100
Flicker with swell	100	100	100	100	100	100
Harmonic	100	100	99.8	100	99.9	100
Interruption	100	99.8	99	98.03	99.5	98.9
Interruption harmonic	97.2	100	100	98.81	98.58	99.4
Normal	100	99.8	96.52	100	98.23	99.89
Notch	100	100	100	100	100	100
Sag	96.8	98	99.38	99.59	98.07	98.79
Sag with harmonic	100	98.8	97.27	100	98.61	99.39
Swell	100	99.8	100	100	100	99.89
Swell with harmonic	99.8	100	100	100	99.89	100
Transient	98	100	100	99.8	98.98	99.9

Table 9: Summary of the models performance between the 1D and the 2D deep CNN

Parameters	1D deep CNN	2D deep CNN X
Accuracy (%)	99.91	99.96
Precision (%)	99.42	99.73
Recall (%)	99.41	99.72
F1-score (%)	99.41	99.72
Time training per epoch (second)	16	30
Model size (MB)	1.24	0.80
File size (MB)	663	128

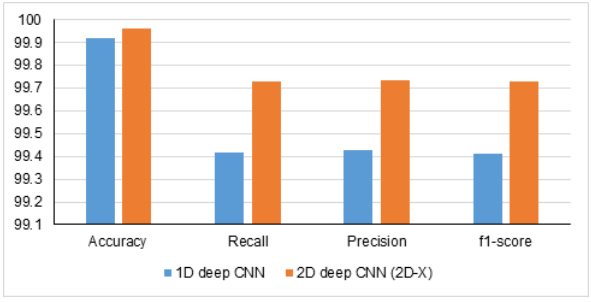


Figure 9: Bar chart of the testing evaluation between the 1D deep CNN and the 2D deep CNN X models

for precision, 99.72% for recall, and 99.72% for f1-score. It indicates that the ability of the 2D deep CNN X model which uses the dataset from our approach to identifying all the relevant disturbances within the dataset is better than the others. In addition, the capability of this model to detect only the disturbances of interest in the dataset is also higher than the previous methods. However, the computation time of our approach is still high with 30 seconds per epoch compared with the other methods. The reason is that the 2D image size resulting from our approach is a large than the previous approaches.

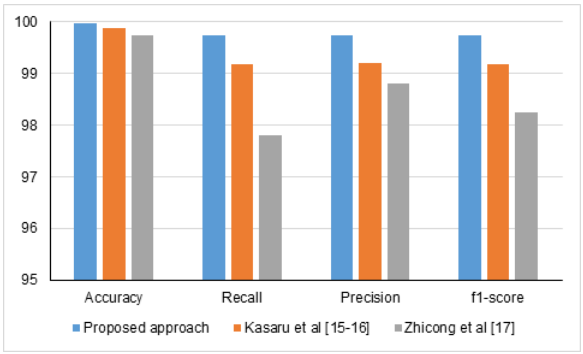


Figure 10: Bar chart of the testing evaluation between our approach and the existing methods

The results of the experiments indicate that the 2D deep CNN model using the 2D image dataset obtained from our approach increases the effectiveness of classification. The signal to image conversion using our approach boosts the 2D deep CNN performance in the PSDs classification, although the computation time is high in a training phase.

Table 10: The testing evaluation between our approach and the existing methods

Parameters	Model with the dataset using the conversion method of		
	Kasaru *et al.* [9, 10]	Zhicong *et al.* [25]	Proposed approach
Accuracy (%)	99.88	99.74	99.96
Precision (%)	99.19	98.80	99.73
Recall (%)	99.18	97.81	99.72
F1-score (%)	99.18	98.25	99.72
Time training per epoch (second)	15	16	30

4 Conclusions

A robust signal to the 2D image conversion and analysis of the PSDs classification based on the 2D deep CNN is presented in this study. In data preprocessing phase, the signal is converted to the 2D grayscale image. The 2D grayscale image preserves the information and sinusoidal waveform of the signal. The conversion results are then utilized as the 2D dataset in the training and testing phase of the model. The experiment's result shows that the accuracy, the recall, and the precision values of the model are 99.96%, 99.72%, and 99.73%, respectively. These result demonstrates that our proposed approach can improve the efficacy of the PSDs classification. In addition, the performance of the proposed approach is better compared to the 1D deep CNN and the previous existing approaches. For a future study, the dataset with noise will be implemented to the 1D and 2D deep CNN model.

Acknowledgments

The authors gratefully acknowledge the helpful comments and suggestions of the reviewers for improving the presentation.

References

[1] A. Aggarwal, N. Das, M. Arora, M. M. Tripathi, "A novel hybrid architecture for classification of power quality disturbances," in *6th International Conference on Control, Decision and Information Technologies (CoDIT'19)*, pp. 1829-1834, 2019.

[2] A. Bagheri, M. H. J. Bollen, I. Y.H. Gu, "Improved characterization of multi-stage voltage dips based on the space phasor model," *Electric Power Systems Research*, vol. 154, pp. 319-328, Jan. 2018.

[3] A. Bagheri, I. Y. H. Gu, M. H. J. Bollen, E. Balouji, "A robust transform-domain deep convolutional network for voltage dip classification," *IEEE Transactions on Power Delivery*, vol. 33, no. 6, pp. 2794-2802, Dec. 2018.

[4] E. Balouji, O. Salor, "Classification of power quality events using deep learning on event images," in *3rd International Conference on Pattern Recognition and Image Analysis (IPRIA'17)*, pp. 216-221, 2017.

[5] S. S. Berutu, Y. C. Chen, "Power quality disturbances classification based on wavelet compression and deep convolutional neural network," in *International Symposium on Computer, Consumer and Control (IS3C'20)*, pp. 327-330, 2020.

[6] K. Cai, W. Cao, L. Aarniovuori, H. Pang, Y. Lin, G. Li, "Classification of power quality disturbances using wigner-ville distribution and deep convolutional neural networks," *IEEE Access*, vol. 7, pp. 119099-119109, Aug. 2019.

[7] IEEE, *IEEE Recommended Practice for Monitoring Electric Power Quality* 2009: c1-81, IEEE 1159.

[8] R. Igual, C. Medrano, F. J. Arcega, G. Mantescu, "Integral mathematical model of power quality disturbances," in *18th International Conference on Harmonics and Quality of Power (ICHQP'18)*, Ljubljana, pp. 1-6, 2018.

[9] S. Karasu, Z. Saraç, "Investigation of power quality disturbances by using 2D discrete orthonormal S-transform, machine learning and multi-objective evolutionary algorithms," *Swarm Evolutionary Computation*, vol. 44, pp. 1060-1072, Feb. 2019.

[10] S. Karasu, Z. Saraç, "Classification of power quality disturbances by 2D-Riesz Transform, multi-objective grey wolf optimizer and machine learning methods," *Digital Signal Processing*, vol. 101, June 2020.

[11] P. Khetarpal, M. M. Tripathi, "A critical and comprehensive review on power quality disturbance detection and classification," *Sustainable Computing: Informatics and Systems*, vol. 28, Dec. 2020.

[12] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D. J. Inman, "1D convolutional neural networks and applications: A survey," *Mechanical Systems and Signal Processing*, vol. 151. Apr. 2021.

[13] H. Liu, F. Hussain, Y. Shen, S. Arif, A. Nazir, M. Abubakar, "Complex power quality disturbances classification via curvelet transform and deep learning," *Electric Power Systems Research*, vol. 163, pp. 1-9. Oct. 2018.

[14] S. K. G. Manikonda, S. Gangwani, S. P. K. Sreckala, J. Santhosh, D. N. Gaonkar, "Power quality event classification using convolutional neural networks on images," in *IEEE 1st International Conference on*

Energy, Systems and Information Processing (ICESIP'19), Chennai, India, pp. 1-5, 2019.

[15] N. Mohan, K. P. Soman, R. Vinayakumar, "Deep power: Deep learning architectures for power quality disturbances classification," in *International Conference on Technological Advancements in Power and Energy (TAP Energy'17)*, pp. 1-6, 2017.

[16] Ş. Öztürk, B. Akdemir, "Cell-type based semantic segmentation of histopathological images using deep convolutional neural networks," *International Journal of Imaging Systems and Technology*, vol. 29, pp. 234-246, Feb. 2019.

[17] Y. Shen, M. Abubakar, H. Liu, F. Hussain, "Power quality disturbance monitoring and classification based on improved PCA and convolution neural network for wind-grid distribution systems," *Energies*, vol. 12, no. 7, pp. 1280, Apr. 2019.

[18] H. Sindi, M. Nour, M. Rawa, Ş. Öztürk, K. Polat, "A novel hybrid deep learning approach including combination of 1D power signals and 2D signal images for power quality disturbance classification," *Expert Systems with Applications*, vol. 174, July 2021.

[19] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168-192, Jan. 2021.

[20] J. Wang, Z. Xu, Y. Che, "Power Quality Disturbance Classification Based on Compressed Sensing and Deep Convolutional Neural Networks," *IEEE Access*, vol. 7, pp. 78336-78346, Jun. 2019.

[21] S. Wang, H. Chen, "A novel deep learning method for the classification of power quality disturbances using deep convolutional neural network," *Applied Energy*, vol. 235, pp. 1126–1140, Feb. 2019.

[22] F. Xiao, T. Lu, M. Wu, Q. Ai, "Maximal overlap discrete wavelet transform and deep learning for robust denoising and detection of power quality disturbance," *IET Generation, Transmission & Distribution*, vol. 14, pp. 140-147, Jan. 2020.

[23] J. Xu, Y. Zhang, D. Miao, "Three-way confusion matrix for classification: A measure driven view," *Information Sciences*, vol. 507, pp. 772-794, Jan. 2020.

[24] H. Xue, A. Chen, D. Zhang, C. Zhang, "A novel deep convolution neural network and spectrogram based microgrid power quality disturbances classification method," in *IEEE Applied Power Electronics Conference and Exposition (APEC'20)*, New Orleans, LA, USA, pp. 2303-2307, 2020.

[25] Z. Zheng, L. Qi, H. Wang, A. Pan, J. Zhou, "Recognition method of voltage sag causes based on two-dimensional transform and deep learning hybrid model," *IET Power Electronics*, vol. 13, pp. 168-177. Jan. 2020.

Biography

Yeong-Chin Chen received the M.S. and Ph.D. degrees in electrical engineering from National Cheng Kung University, Tainan City in 1989 and 1998, respectively. He worked with the Marine Science and Technology Center, Underwater Technology Department, the National Chung Shan Institute of Science and Technology, Taoyuan from 1989 to 2000, as a senior researcher. He conducted research on sonar system design and sound technology. He is currently a professor in computer science and information engineering with Asia University, Taichung. His research interests include acoustical transducer engineering, power signal measurement & analysis, automatic measurement, and software engineering. In recent years, his research has also been focused on smart power meters implemented in the internet of things (IoT) platforms for big-data power saving management applications & Power Quality monitor technology.

Sunneng Sandino Berutu received the B.S. degree in computer science from Immanuel Christian University, Yogyakarta in 2000 and the M.S. degree in information system from University of Diponegoro, Semarang in 2013. He is currently pursuing the Ph.D. degree at Department of Computer Science and Information Engineering, Asia University, Taichung. He is an Information and Technology lecturer at Immanuel Christian University, Yogyakarta, Indonesia. His research interest mainly focused on IoT and power quality.

Long-Chen Hung was born in Kaohsiung, Taiwan, in 1973. He received the M.S. degree from the electrical engineering of I-Shou University, Kaohsiung, in 1998 and the Ph.D. degree from the electrical engineering of Nation Central University, Chungli, in 2005. He is currently an assistant professor in the Department of Electronic Engineering, Lungh-Wa University of Science and Technology. His research interests include deep learning, reinforcement learning, signal processing, control system and AIoT with power quality.

Mariana Syamsudin Mariana Syamsudin is a Ph.d student in Department of Computer Science and Information Engineering Asia University, Taiwan. She earned her bachelor degree in Electrical Engineering from Tanjungpura University, Indonesia and her master degree from Bandung Institute of Technology (ITB), Indonesia. She is an Information and Technology lecturer at Polytechnic State of Pontianak, Indonesia. Her research interest focus on reinforcement learning model.

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:
For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 7,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijns.jalaxy.com.tw or Email to ijns.publishing@gmail.com.