Computer Modeling in
Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2021.015629

ARTICLE

# Steganalysis of Low Embedding Rate CNV-QIM in Speech

**Wanxia Yang[*], Miaoqi Li, Beibei Zhou, Yan Liu, Kenan Liu and Zhiyu Hu**

Mechanical and Electrical Engineering College, Gansu Agricultural University, Lanzhou, 730070, China

[*]Corresponding Author: Wanxia Yang. Email: yanwanxia@163.com

Received: 31 December 2020 Accepted: 27 April 2021

**ABSTRACT**

To address the difficulty of detecting low embedding rate and high-concealment CNV-QIM (complementary neighbor vertices-quantization index modulation) steganography in low bit-rate speech codec, the code-word correlation model based on a BiLSTM (bi-directional long short-term memory) neural network is built to obtain the correlation features of the LPC codewords in speech codec in this paper. Then, softmax is used to classify and effectively detect low embedding rate CNV-QIM steganography in VoIP streams. The experimental results show that for speech steganography of short samples with low embedding rate, the BiLSTM method in this paper has a superior detection accuracy than state-of-the-art methods of the RNN-SM (recurrent neural network-steganalysis model) and SS-QCCN (simplest strong quantization codeword correlation network). At an embedding rate of 20% and a duration of 3 s, the detection accuracy of BiLSTM method reaches 75.7%, which is higher than that of RNN-SM by 11.7%. Furthermore, the average testing time of samples (100% embedding) is 0.3 s, which shows that the method can realize real-time steganography detection of VoIP streams.

**KEYWORDS**

CNV-QIM; steganography; BiLSTM; steganalysis; VoIP; speech

## 1 Introduction

Steganography is an important information security technology that not only hides the content of private information but also its existence to ensure secure transmission [1]. Thus steganography technologies are highly valued by international academic community and military and other security sectors in many countries. As a countermeasure of steganography, steganalysis which is used to analyze whether secret information is hidden in the carrier is also a popular research topic, especially with the emergence of sufficient redundancy digital carriers [2].

The implementation of steganography relies on the carrier. The carriers of steganography could be any kind of data streams, such as images, texts, and audio. In recent years, Voice-over IP (VoIP) streaming media has grown rapidly, and it is difficult to identify due to real-time transmission via the Internet, which is very helpful for hiding secret information. Its massive payloads provide large information hiding capacity. Its multi-dimensional steganography makes detection very difficult. So VoIP becomes one of the best carriers of information hiding.

To reduce band width and improve real-time communication, however, low bit-rate speech codec, such as G.729 and G.723 coders, are used for data compression in VoIP transmission [3,4]. Among the various compression methods, vector quantization (VQ) is the most popular technology in low bit-rate speech coding due to its high compression efficiency and ability to ensure superb voice quality, which also provides a good opportunity for hiding information. Because secret information can be hidden when optimizing the quantized codebook by quantization index modulation (QIM), whose action cannot damage the structure of the carrier with high concealment. Furthermore, the synthesis-by-analysis (ABS) framework applied in low bit-rate speech coding can effectively compensate for the additional distortion caused by the information hidden during residual quantization. These factors make steganalysis for QIM challenging. At present, the complementary neighbor vertices (CNV) algorithm based on graph theory proposed by Xiao et al. [5] is one of state-of-the-art VQ methods for hiding information, and it is the focus of this paper due to its detection difficulty.

Implementation process of the detection method in this paper is illustrated in Fig. 1. Its main characteristics are real-time and fast, the reason is that VoIP is dynamic and real-time, so the steganography detection in VoIP is also real-time, such as the real-time data acquisition. Secondly, if detection is performed offline, it is impractical to cache huge the VoIP data on the network. However, online real-time detection is available. Therefore, a sliding window with window length l and step s is designed to facilitate rapid testing online detection.

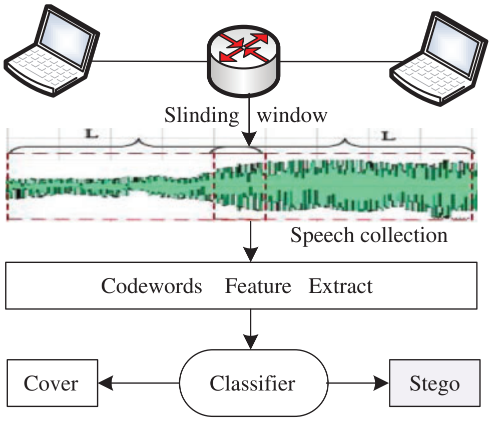


**Figure 1:** Steganalysis process

The remainder of this paper is organized as follows. In Section 2, we discuss the research status of speech steganalysis. In Section 3, a new steganalysis method based on the BiLSTM model is proposed. In Section 4, the detection algorithms designed in Section 3 are simulated, and the experimental results are analyzed. Section 5 concludes our work.

## 2 Related Work

There are many successful methods for detecting steganography in speech coding, especially for LSB steganography. However, in terms of theoretical basis, there are two main categories of features used to steganalysis: the decorrelation of the Mel Frequency Cepstral Coefficient (MFCC) based on the human auditory principle, and the Linear Prediction Coefficient (LPC) and Cepstral Coefficient (LPCC) based on the human vocal mechanism. The former feature is mainly used in LSB steganalysis in high-speed speech coding with low compression rate, while the latter feature in low bit-rate parameter encoder is unsuitable for statistical analysis algorithm. The reason is that the meaning of each parameter in low bit-rate speech codec is not the same. Modifying some parameters by steganography will greatly reduce speech quality and even cause the distortion of speech signal. So the steganography in low bit-rate speech codec is usually implemented when optimizing codebook grouping, and the detection for it is totally different from that of LSB steganography. However, through the analysis of various steganography algorithms used in low bit-rate speech coding, corresponding steganalysis methods are proposed. For example, in [6], histogram flatness, characteristic functions and variance were applied to successfully detect the steganography of the fixed codebook index in the spatial domain. Based on [6,7] added the local extremum of the histogram and a 0, 1 distribution probability difference to improve detection performance. Reference [8] applied Markov correlation of the pulse bit to detect fixed codebook index steganography with a low embedding rate. By introducing the intra-frame and inter-frame codeword correlation network model, Li et al. [9,10] obtained accurate detection for pitch modulation steganography with certain detection difficulty in low bit-rate speech coding. References [11,12] extracted features of the distribution unbalance and loss correlation of code-words caused by steganography and achieved good detection results of QIM steganography of LPC in low bit-rate coders (G.723.1 and G.729). References [13,14] used multiple convolutional network to extract correlation features of the inputted signal to classify and achieved good results, but they lay particular emphasis on the analysis of steganographic speech at different duration.

For the above-mentioned methods, however, the features are manually selected and extracted for steganalysis, which has the following shortcomings. First, the manual selection of features cannot be applied well to the new steganography algorithm; i.e., it is not suitable for discontinuous CNV steganography. Second, voice data is a data stream for which the codewords of successive frames are correlated. However, traditional feature extract algorithms used small samples for classification cannot fully delineate the temporal information between the codeword sequence. Thus, how to fully mine information in the time and space domains of mass samples to improve the accuracy of detection is the crucial problem addressed in this paper. So the recurrent neural network (RNN) model with time memory is chosen to detect CNV-QIM steganography in low bit-rate speech codec in the paper, the model is further explained in the next subsection.

## 3 Speech Steganalysis Based on BiLSTM

### *3.1 BiLSTM Model*

The LSTM model, which was proposed by Graves et al. in 2005 [15], solved the vanishing gradient problem by using gates to selectively memorize information. Thus, it has been successfully applied in speech recognition, natural language processing, etc. [16,17]. For example, Lin et al. [18] used LSTM of RNN to obtain high-accuracy detection for QIM steganography in low bit-rate speech codec (G.729). However, language features, such as the causality between sentences, are logical. The standard LSTM model emphasizes the context information of past language while neglecting the context information of future language. To comprehensively express of semantic relationship between coding parameters in low bit-rate speech and the complete context information of language, the bidirectional LSTM network is designed to descript code-word correlation to get richer and more logical semantic features of speech. Fig. 2 shows a bi-directional RNN unfolded along the time axis.

The composition output of two layers is as follows [19,20]:

$$y = W_{sy}s_t + W_{s'y}s'_t \tag{1}$$

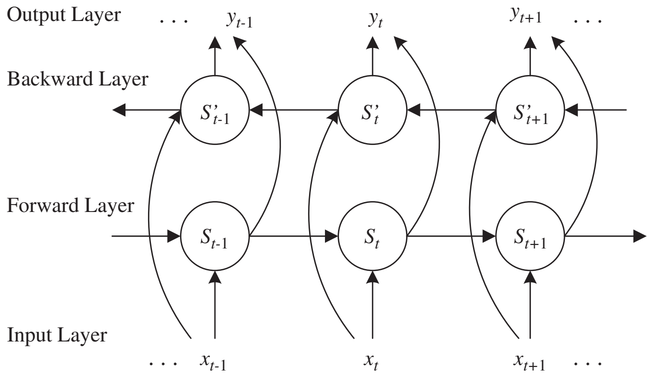


**Figure 2:** BiLSTM model unfolding along the time axis

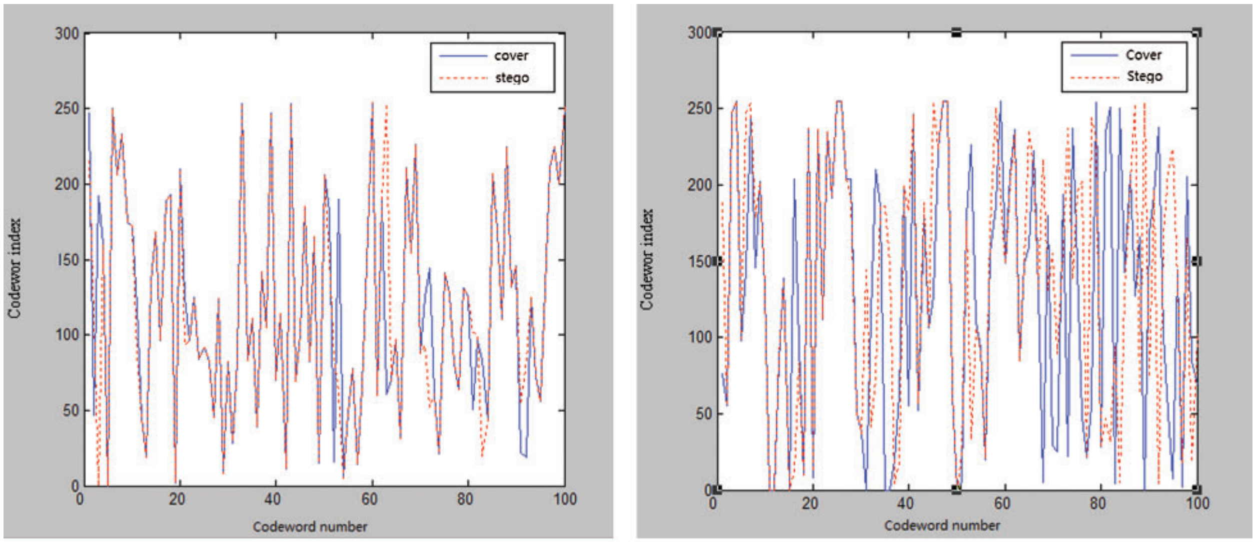
It is obvious that the BiLSTM neural network can better express input information of model compared with LSTM, because it can learn and express the semantic information in speech of both the forward and backward.

### *3.2 Code-Word Correlation Model*

Because speech are short-term stationary signals, they are segmented into short time frames in linear prediction coding, and each frame (10 ms for one frame in G.729 encoder and 30 ms for one frame in G.723 encoder) is analyzed to obtain the optimal LPC, which is then converted to a linear spectral frequency (LSF) quantized by three splitting vectors ($s_1$, $s_2$, $s_3$). Here, each $s_i$ ($i = 1, 2, 3$) corresponds to a codebook $L_i$ ($i = 1,2,3$), whose codeword space is$\{v_i^1, v_i^2, \ldots v_i^{|L_i|}\}$, where $v_i^k$ represents the $k$ codeword of the codebook $L_i$ and $|L_i|$ is the number of codewords in $L_i$. Here, the codewords are one of the main parameters which contain rich voice information. Therefore, there is a strong correlation between the intra-frame and inter-frame codeword and is the main object analyzed in this paper. Which is easy to understand, because speech is the expression of language in the sound. Language is composed of words, and there are rich semantic relations and correlated between the forward and backward words. So the codewords in the coding streams of speech are correlative. In addition, [11,12] used the first-order Markov to model the codeword sequence of speech frame, and applied the state transition probability to quantify the correlation between codewords. However, QIM steganography will alter the correlation of the original codeword distribution while without changing the size of the codeword itself. In this paper, the distribution of the codeword in voice fragments of Chinese male and female is analyzed and the results are shown in Figs. 3a and 3b, respectively. It can be seen that the distribution significant change of the codeword before and after steganography in the experiment. So we design a model based on BiLSTM to delineate distribution change of codeword in codebook $L_i$ caused by steganagraphy. We then use softmax to distinguish cover speech and stego speech. The specific steps of establishing the model are as follows:

The model constructed in this paper consists of two layers of LSTM units (as in Fig. 4). The first layer is the input layer with $N_1$ BiLSTM units defined as $U_1 = \{u_{1,1}, u_{1,2} \cdots u_{1,n1}\}$, and the second layer contains $N_2$ standard LSTM units defined as $U_2 = \{u_{2,1}, u_{2,2} \cdots u_{2,n2}\}$, where $N_1$ and $N_2$ are both 50 determined by experiment in case of considering detection accuracy and efficiency. If the conversion function of the LSTM unit is defined as $f$, then when the input sequence is $\mathrm{I} = \{i_1, i_2 \cdots i_t\}$, the output sequence is $O = \{o_1, o_2 \cdots o_t\}$, as follows:

$$O_{\mathrm{j}} = f(I_{1:j}) \tag{2}$$



(a) (b)

**Figure 3:** Disturbance to the codeword sequence by the CNV-QIM steganography (a) male (b) female

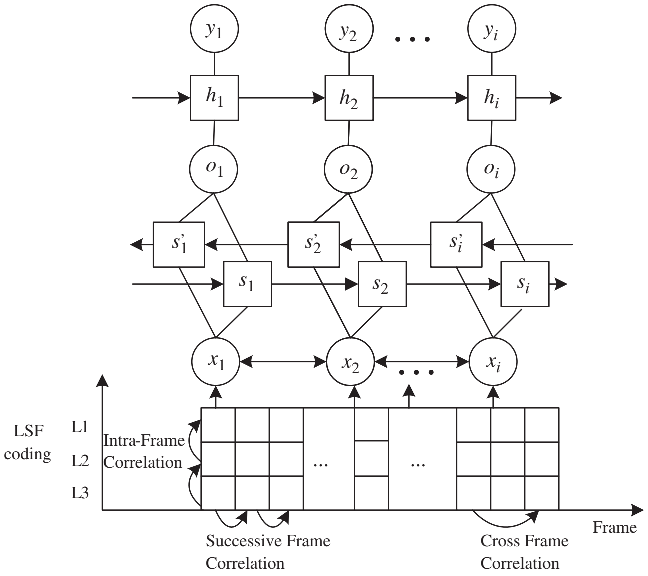


**Figure 4:** Code-word correlation model

Firstly, as Fig. 4, we define all codewords of the $T$ frame in a speech sample as matrix $X$ as follows:

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,T} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,T} \\ x_{3,1} & x_{3,2} & \cdots & x_{3,T} \end{bmatrix} \tag{3}$$

where $x_{1,i}$, $x_{2,i}$ and $x_{3,i}$ represent the LPC coefficients of frame $i$ in codebooks $L_1$, $L_2$ and $L_3$, respectively. For G.729 encoders, $x_{1,i}$, $x_{2,i}$ and $x_{3,i}$ are coded with 7 bits, 5 bits and 5 bits, while codebooks $L_1$, $L_2$ and $L_3$ contain 128, 32 and 32 codewords, respectively. Since steganography changes only $L_1$, $L_2$ and $L_3$, $X$ contains all the information used for steganalysis and serves as the input data of the code-word correlation model.

Secondly, we define forward and backward input weights of codewords $X$ to BiLSTM units in the first layer, these weights reflect how much we should value each codeword, which are expressed in a $3 \times n1$ matrix $A$ and $A'$ as follows, respectively:

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n1} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n1} \\ a_{3,1} & a_{3,2} & \cdots & a_{3,n1} \end{bmatrix} \quad A' = \begin{bmatrix} a'_{1,1} & a'_{1,2} & \cdots & a'_{1,n1} \\ a'_{2,1} & a'_{2,2} & \cdots & a'_{2,n1} \\ a'_{3,1} & a'_{3,2} & \cdots & a'_{3,n1} \end{bmatrix} \tag{4}$$

where for each BiLSTM unit $u_{1,i}$, there are two groups of three input weights $a_{1,i}$, $a_{2,i}$ and $a_{3,i}$ ($a'_{1,i}$, $a'_{2,i}$ and $a'_{3,i}$) multiplied by input codewords $x_{1,i}$, $x_{2,i}$ and $x_{3,i}$, respectively, and the two sets of obtained values are added to form the final input value at each time period. To more specific, the input information value for $u_{1,i}$ at time $t$ is

$$d^1_{i,t} = (a_{1,i} + a'_{1,i})x_{1,t} + (a_{2,i} + a'_{2,i})x_{2,t} + (a_{3,i} + a'_{3,i})x_{3,t} \tag{5}$$

Here, let $A_{1,i} = a_{1,i} + a'_{1,i}$, and $A_{2,i}$ and $A_{3,i}$ are obtained in the same way. Then

$$d^1_{i,t} = A_{1,i}x_{1,t} + A_{2,i}x_{2,t} + A_{3,i}x_{3,t} \tag{6}$$

We define $D^1$ as matrix consisting of $d^1_{i,t}$ together, i.e.,

$$D^1 = \begin{bmatrix} d^1_{1,1} & d^1_{1,2} & \dots & d^1_{1,T} \\ d^1_{2,1} & d^1_{2,2} & \dots & d^1_{2,T} \\ \dots & & & \\ d^1_{n1,1} & d^1_{n1,2} & \dots & d^1_{n1,T} \end{bmatrix} \tag{7}$$

Then, the output of $u_{1,i}$ at t is:

$$o^1_{i,t} = f(D^1_{i,1:t}) = f(a_{i,1}X_{1,1:t} + a_{i,2}X_{2,1:t} + a_{i,3}X_{3,1:t}) \tag{8}$$

We define $O^1$ as matrix gathering all first layer outputs from start to end when input data is $X$, i.e.,

$$O^1 = \begin{bmatrix} o^1_{1,1} & o^1_{1,2} & \dots & o^1_{1,T} \\ o^1_{2,1} & o^1_{2,2} & \dots & o^1_{2,T} \\ \dots & & & \\ o^1_{n1,1} & o^1_{n1,2} & \dots & o^1_{n1,T} \end{bmatrix} \tag{9}$$

At each time, each BiLSTM unit can provide independent output based on the input data and the last state to acquire the preliminary past and future information of codewords. To deeply mine the intrinsic relationships of feature data, we stack another LSTM unit layer to the first layer of the BiLSTM unit in this model, as shown in Fig. 4, and $O^1$ is the input data of the second layer network. For simplicity, the second layer of LSTM has only a forward recurrent neural network layer, but the input and output calculation processes of each unit are similar to those of the first layer unit. Therefore, according to the definition and reasoning process from input to output of the BiLSTM unit, the final output matrix $Y$ (as Eq. (10)) given by formula $Y_j = f(O_j^1)$ of the entire model can be calculated. The matrix $Y$ s also the final correlation feature of all codewords in codebooks $L_1$, $L_2$ and $L_3$, and it can be used for classification.

$$Y = \begin{bmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,T} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,T} \\ \cdots & & & \\ y_{n2,1} & y_{n2,2} & \cdots & y_{n2,T} \end{bmatrix} \tag{10}$$

### 3.3 Feature Classification Model

A linear combination of matrix $Y$ can be used as a classification feature to determine whether the original speech is steganographic, but to state the contribution of each feature for classification, the detection weight (DW) matrix $E(n_2 * T)$ of the model output is defined. Before classification, the output matrix $Y$ is multiplied by DW; then, the classification feature $O^2$ is obtained through linear combinations of their products, as given by Eq. (11).

$$O^2 = \sum_{i=1}^{n2} \sum_{j=1}^{T} Y_{i,j} E_{i,j} \tag{11}$$

In order to facilitate classification, the output $O^2$ is normalized in [0,1] by the sigmoid function to obtain the final output $O^3$.

$$O^3 = S(O^2) \quad = S\left( \sum_{i=1}^{n2} \sum_{j=1}^{T} \mathrm{Y}_{i,j} E_{i,j} \right) \tag{12}$$

We set a detection threshold of 0.5 based on the conventional definition of a threshold in classification. Therefore, the final detection results are determined based on $O^3 \geq 0.5$ (i.e., if $O^3 \geq 0.5$, there is steganography speech; otherwise, it is normal speech).

However, the longer the speech sequence, the faster the growth of the DW matrix, thus slowing the efficiency of training and testing processes. Furthermore, having too many features will increase the possibility of overfitting. To solve this problem, the model is set to output only the feature value of the last unit at T-time; i.e., only $\mathrm{y}_{1:n2,T}$ is a classification feature. Note that the final outputs at end time $T$ count on all outputs at all time steps from the first layer because of LSTM's memorizing ability. In addition, DW is reduced to $N_2$ fixed vectors that do not vary with the length of the input sequence. Therefore, the final output features of the correlation model used to classify are calculated as follows:

$$O^3 = S(O^2) \quad = \mathrm{S}\left( \sum_{\mathrm{i}=1}^{\mathrm{n2}} \mathrm{Y}_{\mathrm{i,T}} \mathrm{E}_{\mathrm{i}} \right) \tag{13}$$

### *3.4 Steganalysis Based on BiLSTM*

The steganalysis process of the method is shown in Fig. 5. The specific procedure is as follows:

(1) The original data is speech samples encoded by G.729a, and the hidden data is obtained by CNV-QIM steganography while encoding. They are segmented by designing sliding windows to acquire more data to train the model.
(2) The features are extracted from each segment of the original and hidden data to constitute the cover and stego feature sets, which are the codeword sequence of speech coding parameters but not the statistical features of the speech signal, where the stego feature set is positive sample 1 and the cover feature set is negative sample 0.
(3) Before training the BiLSTM model using features, it is necessary to preprocess the feature sets into the three-dimensional tensor data for input BiLSTM unit, where the first dimension is the number of samples (batch size), the second is dimensionality of samples, and the third is the length of the sample sequence. Next, Multi-layer LSTM cell model is trained with preprocessed feature data. Finally softmax is used to classify the output data of model.
(4) In the prediction model, the feature data preprocessed in the same way as in Step (3) is input into the trained model for identification, then the model outputs a two-dimensional matrix as a result. The first dimension of the matrix is the number of feature samples, and the second dimension is the mark of seganogrphy or not.

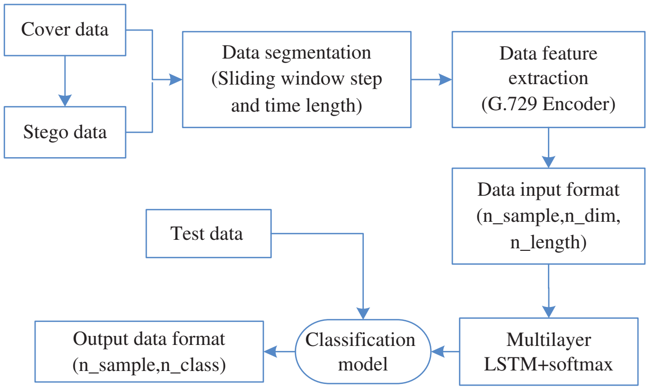

Cover data
Stego data
Data segmentation (Sliding window step and time length)
Data feature extraction (G.729 Encoder)
Data input format (n_sample,n_dim, n_length)
Test data
Output data format (n_sample,n_class)
Classification model
Multilayer LSTM+softmax

**Figure 5:** Steganalysis flow chart

## 4 Experiment and Result Analysis

For an excellent CNV-QIM steganography algorithm based on codebook index in LPC domain, we design steganalysis algorithm based on BiLSTM network and sliding window and carry out experiments (take G.729a coder as an example). The experiment is mainly divided into two processes: the first is the collection and preprocessing of the speech corpus, the second is the training of the model and the comparative experiment.

### *4.1 The Collection and Preprocessing of Corpus*

(1) We collect 98 h of Chinese and English speech samples which are from different male and female speakers in PCM format for forming hybrid speech sample dataset. And these speech samples are filtered through a second-order high-pass filter with a cut-off frequency of 140 Hz.

(2) Filtered speech is encoded by G.729A and segmented into 3, 5 and 10 s samples, respectively, by the sliding window method to form the original speech sample.

(3) The original speech is framed by 80 sample points (10 ms) for short-term LPC analysis. The obtained LPC coefficient is converted to an LSF coefficient for 2-level vector quantization. The first level is a 10-dimensional codebook coded with 7 bits, at the second level of quantification, the 10-dimensional vector is split into two 5-dimensional vectors coded with each 5 bits. The corresponding codebooks are $L_1$, $L_2$ and $L_3$, respectively, and the codewords are extracted from them to form the cover features.

(4) The CNV steganography is performed while encoding the samples in Step (2), and 7 bits, 5 bits and 5 bits codewords in three codebook $L_1$, $L_2$ and $L_3$ are modified in steganography. Thus, only these codewords are extracted for detection to constitute stego features.

(5) In order to verify relationship among embedding rate, sample duration and detection accuracy, the embedding rates of samples in this experiment are 100%, 50% and 20%, respectively, and the sample length of each embedding rate is segmented into 3, 5 and 10 s.

### *4.2 The Training of the Model and Comparative Experiment*

The preprocessed original and steganographic features of samples are mixed. For example, the original feature of 3 s and the steganographic feature of 3 s with 100% embedding rate are blended. Next, 70% of the mixed features (cover and stego) are applied to train the BiLSTM model, and the remaining 30% are used to evaluate the model. Finally, 9000 mixed samples are randomly selected to test the model. Considering the detection accuracy, training time and testing time, we set two layers of LSTM in training model, each of which has 50 units; dropout_W = 0.2, dropout_U = 0.2, loss = 'binary_crossentropy', optimizer = 'adam', and batch_size = 32. We use classification accuracy rate, false positive rate (FP) and false negative rate (FN) to evaluate the performance of model. In the model constructed in this paper, notably, the number of BiLSTM units $N_1$ and LSTM units $N_2$ is particularly important. Therefore, the accuracy, training time and prediction time of the four models with $N_1 = 40$, $N_2 = 50$, $N_1 = 50$, $N_2 = 60$, $N_1 = 60$, $N_2 = 50$, and $N_1 = N_2 = 50$ are designed and firstly compared in the experiment, the results are shown in Tab. 1. As can be seen from Tab. 1, the accuracy of the model with $N_1 = N_2 = 50$ is higher than the other two models, and its training time and prediction time are the shortest. In general, the model with $N_1 = N_2 = 50$ is selected for the subsequent comparative experiments, namely, the method in this paper is compared with the detection methods of RNN-SM [18], IDC (index distribution characteristics) [12] and SS-QCCN [11]. In addition, IDC and SS-QCCN are realized based on SVM, and they cannot detect large samples effectively. Therefore, we randomly extract 4000 samples from mixed features for training IDC and SS-QCCN model and 2000 samples for testing their models. In particular, in order to show the advantages of our detection method for low embedding steganography in short-term speech, the comparative experiments of four methods when the embedding rate of 3%, 5% and 9%, durations of 3 s are conducted. The experimental results of various methods are shown in Tabs. 2–4.

**Table 1:** The analysis of model size

| Embedding rate, Time length | Accuracy, Training time, Prediction time | $N_1 = 40$<br>$N_2 = 50$ | $N_1 = 50$<br>$N_2 = 60$ | $N_1 = 50$<br>$N_2 = 50$ | $N_1 = 60$<br>$N_2 = 50$ |
|---|---|---|---|---|---|
| Er = 20%, Tl = 3 s | Accuracy (%) | 63.2 | 69.9 | 71.9 | 70.8 |
| | Training time (s) | 517 | 562 | 535 | 587 |
| | Predict time (s) | 39.1 | 46.2 | 43.7 | 48.1 |
| Er = 50%, Tl = 3 s | Accuracy (%) | 90.8 | 93.0 | 94.7 | 94.6 |
| | Training time (s) | 526 | 579 | 541 | 581 |
| | Predict time (s) | 40.2 | 46.7 | 41.2 | 47.8 |
| Er = 100%, Tl = 3 s | Accuracy (%) | 97.6 | 98.9 | 98.7 | 99.0 |
| | Training time (s) | 546 | 567 | 551 | 595 |
| | Predict time (s) | 49.3 | 57.2 | 51.3 | 59.8 |

**Table 2:** Detection results of four methods with different embedding rate and 3 s

| Method | Metric | Embedding rate | | | | | |
|---|---|---|---|---|---|---|---|
| | | 3% | 5% | 9% | 20% | 50% | 100% |
| BiLSTM | Acc | **50** | **51.8** | **60.7** | **75.7** | **96.9** | **99.1** |
| | FN | 47.0 | 47.8 | 39.8 | 24.5 | 1.66 | 0.25 |
| | FP | 49.6 | 48.2 | 41.7 | 21.8 | 4.30 | 1.50 |
| RNN-SM | Acc | 48.9 | 50.1 | 51.9 | 67.8 | 90.9 | 99.1 |
| | FN | 48.1 | 47.7 | 46.8 | 27.0 | 8.78 | 0.44 |
| | FP | 50.2 | 49.9 | 48.7 | 34.5 | 8.56 | 1.16 |
| IDC | Acc | 48.3 | 49.5 | 50.5 | 65.8 | 91.3 | 98.7 |
| | FN | 47.5 | 50.6 | 40.6 | 33.0 | 8.40 | 1.40 |
| | FP | 49.9 | 48.4 | 45.4 | 28.7 | 9.10 | 1.30 |
| SS-QCCN | Acc | 49.2 | 51.2 | 57.2 | 67.4 | 97.0 | 99.9 |
| | FN | 56.5 | 48.0 | 42.0 | 28.2 | 2.50 | 0.00 |
| | FP | 45.0 | 49.5 | 41.5 | 37.1 | 3.50 | 0.30 |

It can be seen from Tabs. 2–4 that the detection accuracy of four methods increases with the raising of time length and embedding rate. When the embedding rate is greater than 50% and various durations, the detection accuracy of four methods is almost above 90%. and SS-QCCN has a slight advantage, because it selects a large number of high-dimensional features for detection. However, No matter what length of time, when the embedding rate is less than 50%, the accuracy of our method in this paper is obvious superior than that of the other methods. Especially when the embedding rate is 20% and the time length is 10 s, the detection accuracy of the proposed method reaches 88.8%, which is the only detection result close to 90% among the four methods. It can be explained that the proposed method is more suitable for steganography detection in short-term speech with low embedding rate, which is one of the main purposes of this experiment.

In order to further verify advantages of the proposed method in steganography detection for low embedding speech, we add detection experiments of four methods with embedding rate of 3%, 5% and 9% and time length is 3 s, the results are shown in Tab. 2. From Tab. 2 that the detection effect of method in this paper is the best of the four methods, when the embedding rate is less than 10% and the same time length. Especially when the embedding rate is only 9%, the detection accuracy of our method reaches 60.7% and shows great advantages over other models (Ours (60.7%) *vs.* RNN-SM (51.9%)), existing models represent unsatisfactory detection performance and their detection results are less than 50%. The reason may be that the input layer of our model can fully represent the correlation features of inter-frame and intra-frame codeword while transforming features from low dimension to high dimension. Generally speaking, the detection result of IDC is relatively unsatisfactory among the four methods, when the embedding rate is less than 50%. The possible reason is that it uses SVM method, which is suitable for small sample classification, and its feature dimension is low.

**Table 3:** Detection results of four methods with different embedding rate and 5 s

| Method | Metric | Embedding rate | | |
|---|---|---|---|---|
| | | 20% | 50% | 100% |
| BiLSTM | Acc | **76.4** | **93.9** | 99.7 |
| | FN | 21.1 | 9.91 | 0.17 |
| | FP | 23.7 | 1.65 | 0.34 |
| RNN-SM | Acc | 72.1 | 89.1 | 99.4 |
| | FN | 34.1 | 19.8 | 0.52 |
| | FP | 18.9 | 0.89 | 0.56 |
| IDC | Acc | 68.2 | 90.3 | 99.3 |
| | FN | 32.2 | 12.4 | 1.10 |
| | FP | 29.5 | 7.10 | 0.30 |
| SS-QCCN | Acc | 71.9 | 98.9 | 99.9 |
| | FN | 24.3 | 1.30 | 0.00 |
| | FP | 31.9 | 0.90 | 0.10 |

In order to express the relationship between detection accuracy and embedding rate and time length intuitively, ROC curves at embedding rates of 20% and 50%, durations of 3 s are shown in Figs. 6a and 6b. At the same time, the detection accuracy of proposed method is compared with the other three methods at the embedding rate of 20% and 50%, durations of 3, 5 and 10 s, as illustrated in Fig. 7. It can be clearly seen from the diagram that the consequence is consistent with the result of analysis for table data. In addition, to enable online steganalysis, the time for testing each sample should be as short as possible to improve the detection efficiency. Therefore, the average detecting time of proposed method is compared with the other three methods when the embedding rate is 100% in each time length, and the results are shown in Tab. 5.

**Table 4:** Detection results of four methods with different embedding rate and 10 s

| Method | Metric | Embedding rate | | |
|---|---|---|---|---|
| | | 20% | 50% | 100% |
| BiLSTM | Acc | **88.8** | **93.7** | **99.8** |
| | FN | 5.63 | 5.11 | 0.11 |
| | FP | 15.3 | 6.72 | 0.22 |
| RNN-SM | Acc | 79.9 | 97.5 | 99.8 |
| | FN | 22.2 | 2.99 | 0.10 |
| | FP | 15.5 | 1.66 | 0.17 |
| IDC | Acc | 65.8 | 90.5 | 99.7 |
| | FN | 37.9 | 11.0 | 0.50 |
| | FP | 30.6 | 8.10 | 0.20 |
| SS-QCCN | Acc | 75.5 | 99.9 | 100 |
| | FN | 27.9 | 0.10 | 0.00 |
| | FP | 21.1 | 0.20 | 0.00 |

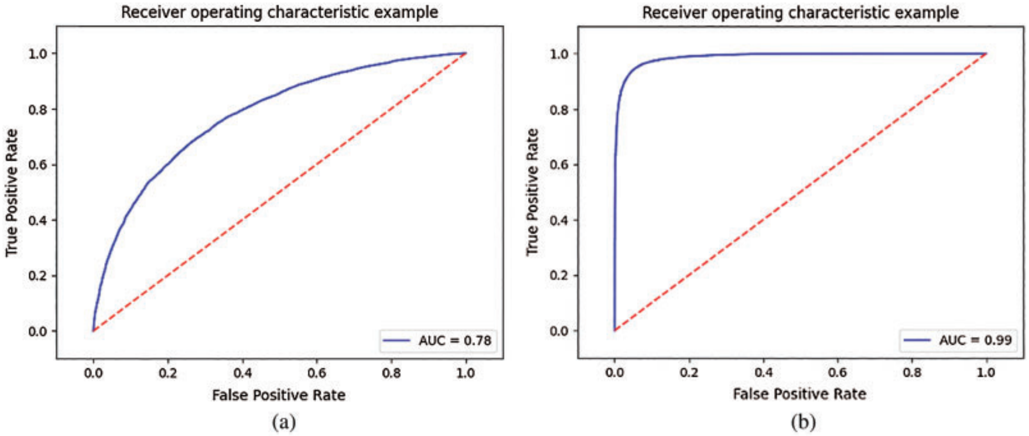


**Figure 6:** ROC curves at different embedding rates and durations of 3 s (a) 20%, 3 s (b) 50%, 3 s

It can be found out from Tab. 5 that although the number of BiLSTM units in this model is more than that of RNN-SM, the average testing time of the model is second only to that of RNN-SM model, and testing time of both models is less than 10% of the sample time length, which can realize real-time steganography detection. However, the overhead of SS-QCCN is distinctly higher than the other three methods due to compute a high dimensional feature vector and need to perform PCA reduction. Combined with the detection accuracy and testing time, the method proposed in this paper is obviously better than other methods.

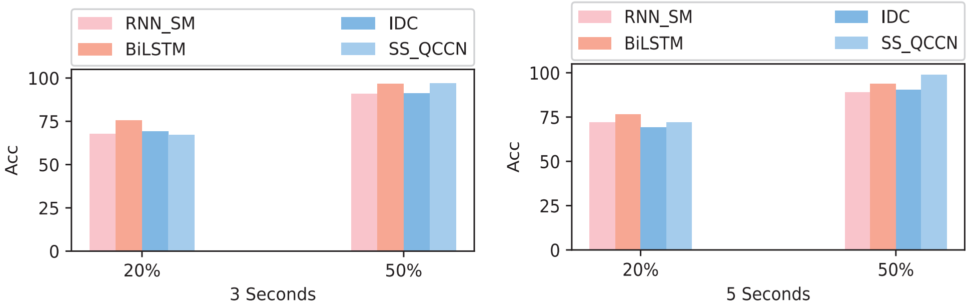


**Figure 7:** Accuracy comparison of four models with embedding rate of 20% and 50% at duration 3, and 5 s

**Table 5:** Testing time comparison at embedding rate 100%

| Method | Testing time | | |
|---|---|---|---|
| | 3 s | 5 s | 10 s |
| BiLSTM | 0.24 | 0.35 | 0.43 |
| RNN-SM | 0.22 | 0.31 | 0.38 |
| IDC | 0.83 | 0.94 | 1.22 |
| SS-QCCN | 1.64 | 1.91 | 2.13 |

## 5 Conclusion and Innovation

To detect CNV-QIM steganography with low embedding rate in low bit-rate speech codec, a steganalysis algorithm based on the BiLSTM neural network model is proposed in this paper, and it achieved good detection results for CNV-QIM steganography of low bit-rate speech codec in VoIP streams. Compared with RNN-SM, IDC and SS-QCCN, the method in this paper has a higher detection accuracy for short samples with low embedding rates. When the embedding rate is 20% and the duration is 3 s, the detection accuracy reaches 75.7%, which is higher than that of the RNN-SM method by up to 11.7%. Especially when the embedding rate is only 9% and durations of 3 s, the detection accuracy of our method reaches 60.7% and shows great advantages over other models. Furthermore, in terms of detection efficiency, the average testing time of samples (100% embedding) is 0.3 s, which shows that the method can realize real-time steganography detection of VoIP streams.

There are two innovations in this paper: (1) Accurate steganography detection in low bit rate speech with low embedding rate is realized by building code-word correlation model using BiLSTM. (2) Real-time detection for timely voice is achieved.

**Acknowledgement:** This work was supported by grants from the National Natural Science Foundation of China (No. 61862002). The authors would like to thank the anonymous reviewers for their valuable suggestions.

**Funding Statement:** This research was supported by the National Natural Science Foundation of China (No. 61862002).

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

## References

1. Tian, H., Sun, J., Chang, C. C., Qin, J., Chen, Y. (2017). Hiding information into voice-over-ip streams using adaptive bitrate modulation. *IEEE Communications Letters, 21(4),* 749–752. DOI 10.1109/LCOMM.2017.2659718.
2. Qin, C., Hu, Y. C. (2016). Reversible data hiding in vq index table with lossless coding and adaptive switching mechanism. *Signal Process, 29,* 48–55. DOI 10.1016/j.sigpro.2016.05.032.
3. Huang, Y. F., Liu, C. H., Tang, S. Y. (2012). Steganography integration into a low-bit rate speech codec. *IEEE Transactions on Information Forensics and Security, 7(6),* 1865–1875. DOI 10.1109/TIFS.2012.2218599.
4. Tian, H., Sun, J., Chang, C. C., Huang, Y., Chen, Y. (2018). Detecting bitrate modulation-based covert voice-over-ip communication. *IEEE Communications Letters, 22(6),* 1196–1199. DOI 10.1109/LCOMM.2018.2822804.
5. Xiao, B., Huang, Y. F., Tang, S. Y. (2008). An approach to information hiding in low bit-rate speech stream. *IEEE Global Telecommunications Conference*, pp. 1–5. New Orleans, LA, USA.
6. Yan, D. Q., Wang, R. D. (2014). Detection of m-p3stego exploiting recompression calibration-based feature. *Multimedia Tools and Applications, 72(1),* 865–878. DOI 10.1007/s11042-013-1406-z.
7. Guo, H., Yan, D., Wang, R., Wang, Z., Wang, L. et al. (2015). Mp3 steganalysis based on difference statistics. *Computer Engineering and Applications, 51(7),* 88–92.
8. Tian, H., Wu, Y., Huang, Y., Liu, J., Chen, Y. et al. (2015). Steganalysis of low bit-rate speech based on statistic characteristics of pulse positions. *10th International Conference on Availability, Reliability and Security*, pp. 455–460. Toulouse, France, August.
9. Li, S. B., Jia, Y. Z., Fu, J. Y., Dai, Q. X. (2014). Detection of pitch modulation information hiding based on code-book correlation network. *Chinese Journal of Computers, 37(10),* 2107–2116.
10. Li, S. B., Huang, Y. F., Lu, J. C. (2013). Detection of QIM steganography in low bit-rate speech codec based on statistical models and SVM. *Chinese Journal of Computers, 36(6),* 1168–1176. DOI 10.3724/SP.J.1016.2013.01168.
11. Li, S. B., Jia, Y. Z., Kuo, C. C. (2017). Steganalysis of QIM steganography in low-bit-rate speech signals. *IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(5),* 1011–1022. DOI 10.1109/TASLP.2017.2676356.
12. Li, S. B., Tao, H. Z., Huang, Y. F. (2012). Detection of quantization index modulation steganography in G. 723.1 bit stream based on quantization index sequence analysis. *Journal of Zhejiang University-SCIENCE C, 13(8),* 624–634. DOI 10.1631/jzus.C1100374.
13. Yang, Z. L., Guo, X. Q., Chen, Z. M., Huang, Y. F., Zhang, Y. J. (2018). RNN-stega: Linguistic steganography based on recurrent neural networks. *IEEE Transactions on Information Forensics and Security, 14(5),* 1280–1295. DOI 10.1109/TIFS.2018.2871746.
14. Yang, Z. L., Zhang, S. Y., Hu, Y. T., Hu, Z. W., Huang, Y. F. (2020). VAE-Stega: Linguistic steganography based on variational auto-encoder. *IEEE Transactions on Information Forensics and Security, 16,* 880–895. DOI 10.1109/TIFS.10206.
15. Graves, A., Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks, 18(5),* 602–610. DOI 10.1016/j.neunet.2005.06.042.
16. Chen, C. T., Zhuo, R., Ren, J. T. (2019). Gated recurrent neural network with sentimental relations for sentiment classification. *Information Sciences, 502,* 268–278. DOI 10.1016/j.ins.2019.06.050.

17. Jorge, C. Z., Alejandro, H. T., Enrique, V. (2019). Handwritten music recognition for mensural notation with convolutional recurrent neural networks. *Pattern Recognition Letters, 128,* 115–121. DOI 10.1016/j.patrec.2019.08.021.
18. Lin, Z. N., Huang, Y. F. (2018). Rnn-sm: Fast steganalysis of voip streams using recurrent neural network. *IEEE Transactions on Information Forensics and Security, 13(7),* 1854–1868. DOI 10.1109/TIFS.2018.2806741.
19. Yang, Z., Huang, Y., Jiang, Y., Sun, Y., Zhang, Y. J. et al. (2018). Clinical assistant diagnosis for electronic medical record based on convolutional neural network. *Scientific Reports, 8(1),* 6329. DOI 10.1038/s41598-018-24389-w.
20. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research, 15(1),* 1929–1958.