Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.036419
Article

Tech Science Press

Using Recurrent Neural Network Structure and Multi-Head Attention with Convolution for Fraudulent Phone Text Recognition

Junjie Zhou, Hongkui Xu*, Zifeng Zhang, Jiangkun Lu, Wentao Guo and Zhenye Li

School of Information and Electrical Engineering, Shandong Jianzhu University, Jinan, 250000, China
*Corresponding Author: Hongkui Xu. Email: xhkui2009@163.com

Received: 29 September 2022; Accepted: 14 December 2022

Abstract: Fraud cases have been a risk in society and people's property security has been greatly threatened. In recent studies, many promising algorithms have been developed for social media offensive text recognition as well as sentiment analysis. These algorithms are also suitable for fraudulent phone text recognition. Compared to these tasks, the semantics of fraudulent words are more complex and more difficult to distinguish. Recurrent Neural Networks (RNN), the variants of RNN, Convolutional Neural Networks (CNN), and hybrid neural networks to extract text features are used by most text classification research. However, a single network or a simple network combination cannot obtain rich characteristic knowledge of fraudulent phone texts relatively. Therefore, a new model is proposed in this paper. In the fraudulent phone text, the knowledge that can be learned by the model includes the sequence structure of sentences, the correlation between words, the correlation of contextual semantics, the feature of keywords in sentences, etc. The new model combines a bidirectional Long-Short Term Memory Neural Network (BiLSTM) or a bidirectional Gate Recurrent United (BiGRU) and a Multi-Head attention mechanism module with convolution. A normalization layer is added after the output of the final hidden layer. BiLSTM or BiGRU is used to build the encoding and decoding layer. Multi-head attention mechanism module with convolution (MHAC) enhances the ability of the model to learn global interaction information and multi-granularity local interaction information in fraudulent sentences. A fraudulent phone text dataset is produced by us in this paper. The THUCNews data sets and fraudulent phone text data sets are used in experiments. Experiment results show that compared with the baseline model, the proposed model (LMHACL) has the best experiment results in terms of Accuracy, Precision, Recall, and F1 score on the two data sets. And the performance indexes on fraudulent phone text data sets are all above 0.94.

Keywords: BiLSTM; BiGRU; multi-head attention mechanism; CNN

1 Introduction

In recent years, the new crime represented by telecom fraud is developing rapidly. Phone fraud, as one of the types, has been a risk in society. People's lives and property have been threatened seriously. According to relevant data, the top ten fraud methods [1] are part-time fraud, counterfeiting fraud, pornography fraud, free fraud, number theft fraud, transaction fraud, rebate fraud, low price inducement fraud, financial credit fraud, and friend-making fraud. These methods are cross-used in fraudulent words, making them difficult to prevent.

In the field of fraudulent phone recognition, traditional machine learning algorithms [2] are used by some researchers to identify fraudulent calls by extracting the difference between fraudulent calls and normal calls in terms of call frequency and call time. There are also researchers using deep learning methods [3] to replace artificial features with automatic features, which improves the recognition accuracy of fraudulent calls and ensures timeliness. Ying et al. [4] proposed an efficient fraud phone detection framework based on parallel graph mining, which can automatically label fraudulent phone numbers to generate phone number trust values. However, the above methods rely on a large amount of user behavior data and are unable to detect the semantics of the phone content, which results in the inability to identify a new phone number.

Classifying the text content of a fraudulent phone can recognize a fraudulent phone in a timely and effective. Fraud semantics are sometimes extremely difficult to distinguish only by people. The fraudulent phone text consists of some sentences that contain fraudulent semantics. The meaning of these sentences is related to the sequence structure of the sentences, the correlation between words, the correlation of contextual semantics, and the feature of keywords in sentences. And this knowledge can be learned by deep learning models to determine the possibility of fraudulent phone texts. Text classification [5] is a method of identifying text categories in deep learning. Classification rules are formulated by model learning the semantic content of the text. The process can be regarded as a functional relationship. According to the functional relationship, fraudulent texts are classified as a fraud category. In recent studies, RNN, the variants of RNN, CNN, and hybrid neural networks to extract text features are used by most text classification research. For example, many promising algorithms have been developed for social media offensive text recognition as well as sentiment analysis. However, a single network or a simple network combination cannot obtain rich characteristic knowledge of fraudulent phone texts relatively. Therefore, a classification model for fraudulent phone text is proposed in this paper, which adopts the method of text classification to identify fraudulent phone calls on the user side.

The overall contributions of this paper are summarized as follows:

(1) BiLSTM or BiGRU is used as an encoding-decoding structure to extract the information of sequence structure and contextual semantic correlation of sentences in fraudulent phone text, thereby enhancing the ability of the model to learn fraud knowledge. In the comparison experiment, BiGRU had higher accuracy compared with BiLSTM in our model and we finally used BiGRU as the encoding decoding structure.

(2) An enhancement module MHAC is added to the structure of encoding and decoding to mine deep fraud semantic knowledge. The model can learn the global interaction information and multi-granularity local interaction information of the fraudulent phone text through this module, which greatly enriches learning content.

(3) To stabilize the output vector of the hidden layer and prevent the model from overfitting, a normalization layer is added after the decoding layer of the model, which improves the

performance of the model. And an ablation experiment was done to confirm the effect of this layer.

(4) A series of experiments are conducted on a public dataset and the fraudulent phone text dataset is constructed in this paper. Experiment results show that compared with the baseline model, LMHACL has the best experiment results in terms of Accuracy, Precision, Recall, and F1 score on the two data sets. And the performance indexes on fraudulent phone text data sets are all above 0.94.

2 Related Work

Essentially, fraudulent phone text is some textual content that contains scam semantics. In the text classification method, the knowledge of different aspects of the text can be learned through the neural network, to identify the fraudulent text. Text classification methods are mainly divided into two categories: methods based on machine learning and methods based on deep learning. Methods based on machine learning rely more on hand-crafted features and are lacking in both efficiency and feature representation. Methods based on deep learning have gradually become mainstream due to their superiority in efficiency and accuracy. The research on text classification [6] generally uses word embeddings to represent text content and neural networks to extract text features.

Using word embedding [7] to represent fraudulent phone text, its semantic information can be accurately expressed. Word2Vec [8], and Glove [9] are representative word embeddings in text classification tasks. Word2Vec is a distributed representation method. Compared with the early one-hot encoding and term frequency-inverse document frequency (TF-IDF) methods, this method is richer and more accurate in expressing text semantics, and it solves the problems such as the curse of dimensionality and the problem of data sparsity [10] in early research. In classification experiments on online news and Twitter network data, the effectiveness of adding Word2Vec to the model was confirmed by Jang et al. [11].

The time-series features of a text can be recorded by RNN, and many researchers use it to process sentiment analysis tasks. However, it has the problems of gradient explosion and gradient disappearance. Therefore, some researchers have improved it. Some scholars proposed a Long-Short Term Memory Neural Network (LSTM) [12], which solved this problem with three unique gate structures. Some scholars have also proposed a Gated Recurrent Unit (GRU) [13], which simplifies the network structure while solving the problem and shortens the training time. Fig. 1a is the LSTM structure, and the relevant calculation is as formula (1), and Fig. 1b is the GRU structure, and the relevant calculation is as formula (2).

$$f_t = \sigma\left(W_f \cdot [H_{t-1}, T_t] + b_f\right)$$

$$I_t = \sigma\left(W_i \cdot [H_{t-1}, T_t] + b_i\right)$$

$$O_t = \sigma\left(W_o \cdot [H_{t-1}, T_t] + b_o\right)$$

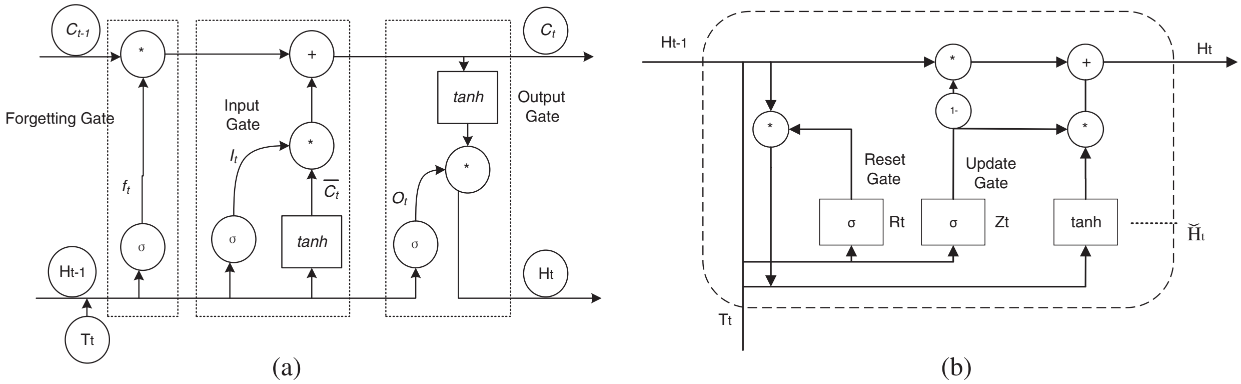
$$H_t = O_t * \tanh\left(C_t\right)$$

$$C_t = f_t * C_{t-1} + I_t * \overline{C_t}$$

$$\overline{C_t} = \tanh\left(W_c [H_{t-1}, T_t] + b_c\right) \tag{1}$$

where f_t represents the forget gate, I_t represents the input gate, and O_t represents the output gate. σ represents the sigmoid activation function, H_{t-1} is the hidden state at time $t-1$, T_t is the current input,

C_t is the current state, and $\overline{C_t}$ is the current memory state. W_f, W_i, and W_o represent the weight of each gate respectively. b_f, b_i, and b_o represent the bias of the three gates respectively.



(a) (b)

Figure 1: Structure of LSTM/GRU. (a) LSTM; (b) GRU

The output of the gate is controlled by σ and outputs a vector of real numbers between 0 and 1. When the output of the gate is 0, any vector is multiplied by it and gives you the 0 vector, which is equivalent to passing no information. When the output is 1, any vector multiplied by it does not change and all information is passed. In summary, the gate has many advantages. (1) It can save information long ago. (2) It can avoid insignificant information entering the memory. (3) It can control the influence of long-term memory on current output.

$$Z_t = \sigma\left(W_j T_t + U_j H_{t-1} + b_z\right)$$

$$R_t = \sigma\left(W_k T_t + U_k H_{t-1} + b_r\right)$$

$$\widetilde{H}_t = \tanh\left(W_{th} T_t + \left(R_t \odot H_{t-1}\right) U_{th} + b_h\right)$$

$$H_t = \left(1 - Z_t\right) \odot H_{t-1} + Z_t * \widetilde{H}_t \tag{2}$$

where Z_t and R_t represent the update gate and reset gate respectively, $\widetilde{H}_t$ represents the candidate hidden state, $\odot$ represents the inner product of vectors, W_j, U_j, W_k, U_k, W_{th}, and U_{th} are the weight parameters, and H_t is the state output of hidden layer at time t.

The GRU combines the forget gate and the input gate to form the update gate. Updating the amount of information saved from the previous time to the current time. The smaller value means that more past information is stored. The function of the reset gate is to control the amount of ignoring state information at the previous moment. The smaller the value, the greater the amount of ignoring information. Compared with LSTM, GRU has one less gate function and the number of parameters is reduced.

Research shows that contextual information of text can be better learned by the bidirectional network structure. For example, Xu et al. [14] adopted BiLSTM to achieve high-performance scores in the sentiment analysis task of online comment texts. In the text classification task of the financial field, Leng et al. [15] compared the classification performance improvement of BiGRU and BiLSTM through experiments. The contextual relevance of the text and long-distance dependence are extracted by the forward hidden layer and the backward hidden layer in each time step of BiLSTM and BiGRU, which is very helpful for learning fraud semantic information by LMHACL.

Local interaction information of the text can be captured by CNN [16]. In the fraud text, phrase-level semantics of different lengths in each sentence are recorded by these local interactions, which are also important for enriching fraud features. CNN has achieved good results in the classification task. Many studies combine the advantages of RNN and CNN networks to integrate sequence information and local interaction information to improve text classification performance. Huan [17] et al. proposed a model based on CNN and LSTM to extract shallow local semantic features and deep global semantic features of complex semantic texts. To highlight the role of key information in the text, an attention mechanism has been added to the model by some studies. Bao et al. [18] combined CNN and LSTM to obtain local and long-distance contextual information on the steganographic text, and the attention mechanism was used to focus on important clues in text sentences and achieved good results in steganographic text analysis tasks. Liu et al. [19] combined the advantages of BiGRU, attention mechanism, and CNN according to the characteristics of the Chinese question, and the features were effectively extracted. The word context information is learned by the model, and the problem of TextCNN losing positional features is solved at the same time. With the continuous development of the attention mechanism, the multi-head attention mechanism was born. In 2017, Google created the Transformer model based on the multi-head attention mechanism.

Transformer [20] uses the self-attention mechanism instead of CNN and RNN, which has the ability of parallel computing and extraction of long-distance features. Its structure is composed of the multi-head attention mechanism and the feedforward neural network. It is a typical seq2seq structure with strong feature extraction capabilities. Tezgider et al. [21] constructed a bidirectional Transformer structure, and the bidirectional encoding information of text can be learned from this structure. The experimental results show that the text features extracted by the bidirectional Transformer have a strong effect on the classification results. In addition, many excellent models have been built by some researchers with the powerful ability of Transformer, such as BERT [22], ERNIE [23], ALBERT [24], etc. These models not only perform well in the field of text classification but also in other fields of natural language processing. Although Transformer is good at capturing global contextual information, its ability to extract fine-grained local features is poor. In the field of speech recognition, Gulati et al. [25] skillfully combined the Transformer and CNN, combining the ability of the Transformer to capture content-based global interaction and the ability of CNN to extract fine-grained local interaction features, a structure called Conformer is built. The structural idea of Conformer is used for reference in this paper. A multi-head attention module with convolution, which has a similar structure to Conformer is constructed. And the global interaction information and multi-granularity local interaction information in the fraudulent phone text is learned.

A fraud phone text classification model is proposed in this paper, which combines BiGRU, BiLSTM, feedforward neural network, multi-head attention mechanism, and CNN. A fraudulent phone text dataset is built in this paper and uses Word2Vec to represent fraudulent phone text. BiGRU or BiLSTM is used as a codec to learn the correlation of sequence-structure information and contextual semantics of sentences in fraudulent phone text. MHAC module is used to enhance the ability of the model to learn global and multi-granularity local interaction information in fraudulent statements. A normalization layer is added after the output of decoding to enhance the fitting ability of the model. In our model, multi-level fraud semantic knowledge is learned, so that fraud phone text can be classified with high performance.

3 Methodology

The semantics of fraud phone texts are more complex and difficult to distinguish than those of general texts. Therefore, multi-level semantic knowledge of text needs to be learned by the fraudulent phone text classification model. The overall structure of the model is shown in Fig. 2, which is mainly divided into 5 parts, Embedding Layer, Encoding Layer, Enhancement Module with Multi-Head Attention and Convolution, Decoding Layer, and Classification Layer.

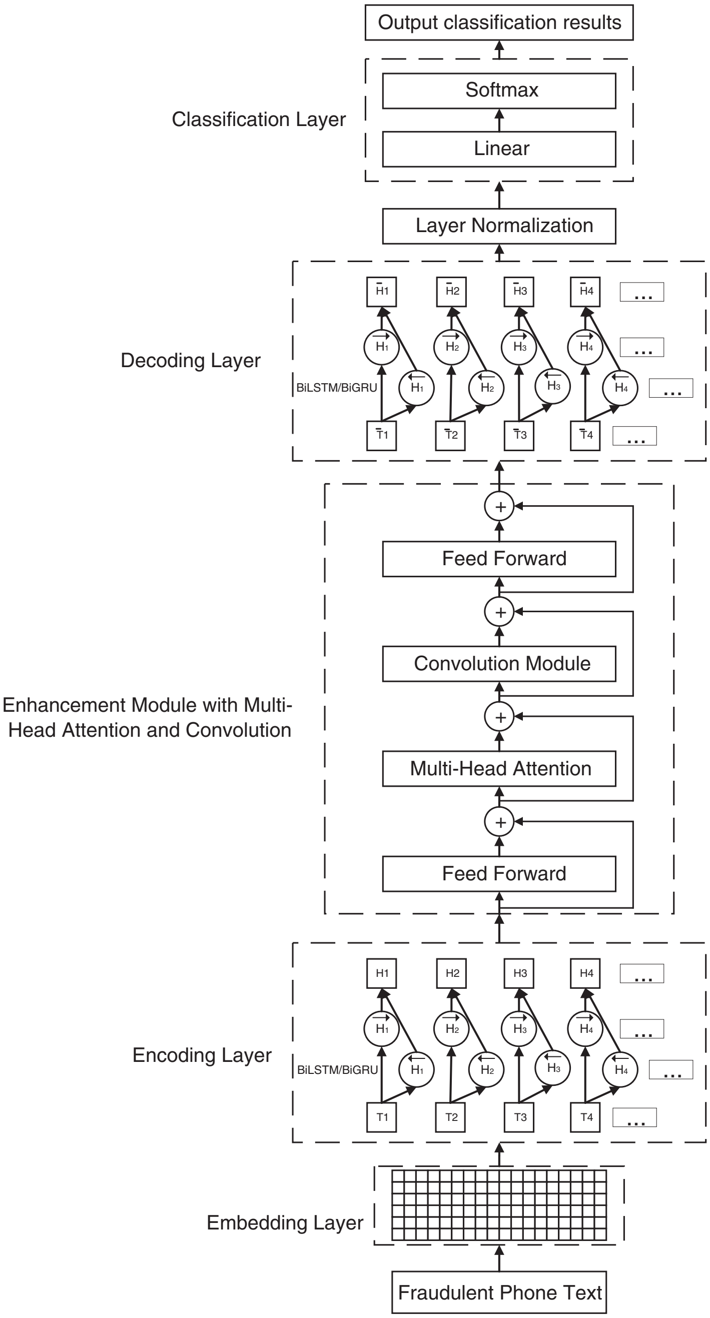


Figure 2: Structure of fraudulent phone text classification model

3.1 Embedding Layer and Encoding Layer

Word2Vec is used as the word embedding vector for our model. A sentence containing m words in fraud phone text can be expressed as $X = [X_1, X_2, X_3, \ldots, X_m]$. The word sequence X is converted into word vector T through the Embedding layer. $T = [T_1, T_2, T_3, \ldots, T_m]$, $T_i \in R^d$ represents the i_{th} word vector of dimension d, $T \in R^{m \times d}$ represents the sentence vector.

The three-gate structure is used by LSTM to record the time series information of fraudulent phone text, while the structure was simplified by GRU and two gates were used. BiGRU or BiLSTM is used by the Encoding layer as the encoder, and the information in the front and rear directions of the text is received by the bidirectional structural model so that the contextual relevance and text sequence information of the fraudulent phone text is obtained. The calculation formulas are as follows.

$$\overrightarrow{H_L} = \overrightarrow{LSTM}(T)$$

$$\overleftarrow{H_L} = \overleftarrow{LSTM}(T)$$

$$\overrightarrow{H_G} = \overrightarrow{GRU}(T)$$

$$\overleftarrow{H_G} = \overleftarrow{GRU}(T) \tag{3}$$

$$H_L = \left[\overrightarrow{H_L}, \overleftarrow{H_L}\right]$$

$$H_G = \left[\overrightarrow{H_G}, \overleftarrow{H_G}\right] \tag{4}$$

where $\overrightarrow{H_L}$ and $\overrightarrow{H_G}$ represent the forward hidden layer outputs of LSTM and GRU, respectively. $\overleftarrow{H_L}$ and $\overleftarrow{H_G}$ represent the backward hidden layer outputs of LSTM and GRU, respectively. H_L and H_G are the outputs of the Encoding layer.

3.2 Enhancement Module with Multi-Head Attention Mechanism and Convolution

If the fraudulent phone text is understood separately, sometimes the intention of fraud is not reflected in the sentence, so the semantics of the text needs to be mined at multiple levels. An enhancement module MHAC is proposed in this paper, the advantages of the feedforward neural network, multi-head attention mechanism, and convolutional neural network are combined in this module, and residual links are introduced at the same time. The global interaction features and multi-granularity local interaction features of fraudulent phone text are learned by it. This module is shown in Fig. 3.

3.2.1 Multi-Head Attention Mechanism

The entire fraud sentence is integrated into each word in the sentence by the multi-head attention mechanism so that each word contains extremely rich global information, which is also conducive to the semantic disambiguation of polysemy. For example, “Daily salary is 200 yuan, but a deposit of 200 yuan is required”. In this sentence, “200 yuan” appears twice, but from a global perspective, the global information contained in the sentence is different. Another example is, “Just now the post office contacted me and said that an emergency email was sent to me by the public security bureau, but I didn’t receive it. They asked me to call and ask about the situation”. The word “post office” is analyzed by the global interaction of the multi-head attention mechanism, and the degree of association between “they” and the word “post office” is calculated to be the highest. Although the two words are far apart, this relationship is still recorded by the learning process of global interaction.

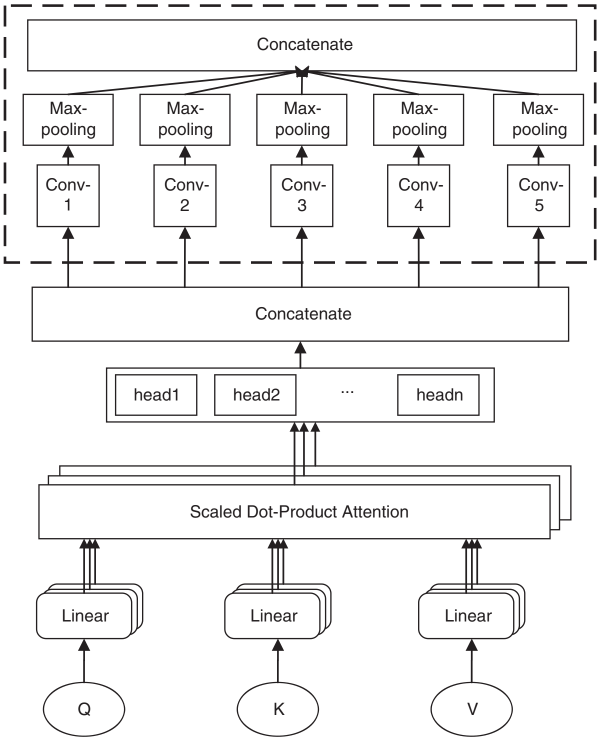


Figure 3: Structure of MHAC

The feedforward neural network is a sub-network in the module. The network consists of two layers of linear functions and the ReLU activation function. At the same time, Dropout is added, the nonlinear ability of the model is enhanced, and the network is standardized. This structure is used in both Transformer and Conformer. The calculation formula is formula (5).

$$\mathrm{FFN1} = \mathrm{Max}(0, \mathrm{HW}_1 + \mathrm{b}_1)\mathrm{W}_2 + \mathrm{b}_2 \tag{5}$$

where W_1, W_2, b_1, and b_2 are training parameters, H is the output of the Encoding Layer.

The correlation information between any word and other words in the fraud sentence from multiple angles is recorded by the multi-head attention mechanism. We can think of these angles as referential relations, state relations or parallel relations, etc. In the above example, "post office" and "they" are referential relationships, "emergency" and "email" are state relationships, and "public security bureau" and "post office" are parallel relationships. Therefore, in each angle, the degree of relationship is different, and the weight is different. The multi-head attention mechanism is shown in Fig. 3. Three vectors Q, K, and V with different information are generated according to the input vector, as shown in formula (6).

$$\mathrm{Q} = (\mathrm{FFN1} \oplus \mathrm{H})\mathrm{W}^{\mathrm{Q}}$$

$$\mathrm{K} = (\mathrm{FFN1} \oplus \mathrm{H})\mathrm{W}^{\mathrm{K}}$$

$$\mathrm{V} = (\mathrm{FFN1} \oplus \mathrm{H})\mathrm{W}^{\mathrm{V}} \tag{6}$$

where Q is the query vector, K is the queried vector, and V is the actual feature vector. W^Q, W^K, and W^V are random initialization matrices. FFN1 ⊕ H represents the residual connection to prevent information loss during transmission. ⊕ represents bitwise addition.

According to formula (7), the vector N of the comprehensive feature under a single angle is calculated.

$$N = \mathrm{OneAtt}(Q, K, V) = \mathrm{Softmax}\left(\frac{QK^T}{\sqrt{d_m}}\right)V \tag{7}$$

where d_m is the vector dimension, and N is the global interaction information contained in the feature vector of each word at the current angle.

Multiple sets of W^Q, W^K, and W^V matrices are initialized, and the global interaction information at different angles can be obtained. The global interaction of each angle is combined to form the global interaction of the multi-head attention mechanism, as shown in the formula (8).

$$\mathrm{MATT}\,(Q, K, V) = \mathrm{concat}(\mathrm{head}_1, \mathrm{head}_2, \ldots, \mathrm{head}_n)\,W^0 \tag{8}$$

where $\mathrm{head}_i = \mathrm{OneAtt}(QW_i^Q, KW_i^K, VW_i^V)$, head_i represent the global interaction information of the fraudulent text at the i_{th} angle, with a total of n angles. W_i^Q, $W_i^K \in R^{dm\times dk}$, $W_i^V \in R^{dm\times dv}$, $W^0 \in R^{hd\times dm}$.

3.2.2 Convolution Module

Multi-grained local interaction information in scam text can be recorded by the convolution module. A single-word vector is difficult to express the meaning of the whole sentence. After adding the convolution module, the word vector can be convolved to obtain a phrase-level vector. When multiple scales of convolution kernels are used, phrase-level vectors of different lengths can be output. These vectors are fused, and the multi-granularity local interaction information in the fraud text can be recorded. The convolution module is shown in Fig. 3.

The input vector MATT is convolved by the kernel E of different convolution scales. The convolution scale represents the number of words that can be covered, also known as the convolution width. The calculation is shown in formula (9).

$$TC_i = \tanh\left(\langle E, (\mathrm{MATT} \oplus \mathrm{MATT}_F)\rangle + b\right) \tag{9}$$

where $<\cdot>$ represents the convolution calculation, $\mathrm{MATT}\oplus\mathrm{MATT}_F$ represents the residual connection and $\mathrm{MATT}_F = \mathrm{FFN1}\oplus$. TC_i is the feature vector after convolution. $\mathrm{MATT}\in R^{m\times hd}$, $E\in R^{m\times h}$, h is the width of the convolution kernel, and m is the dimension of the input vector.

Maximum pooling is performed after convolution, as in formula (10).

$$MTC_i = \max_i TC_i \tag{10}$$

when using s different convolution kernels, we can get s outputs after passing the convolution module. These outputs are stitched together to obtain multi-granularity local interaction feature vectors. The calculation is shown in formula (11).

$$MTC = [MTC_1, MTC_2, \ldots, MTC_s] \tag{11}$$

where $E_1\in R^{m\times h1}$, $E_2\in R^{m\times h2}$, …, $E_s\in R^{m\times hs}$, $MTC\in R^{m\times hd}$.

The vector from the previous layer is mapped by the feedforward neural network so that the global interaction information and the local interaction information are fused. The calculation is shown in formula (12).

$$\mathrm{FFN2} = \max\left(0, (MTC \oplus MTC_M)W_3 + b_3\right)W_4 + b_4 \tag{12}$$

where W_3, W_4, b_3, b_4 represent training parameters, MTC is the output of the convolution block, $\mathrm{MTC} \oplus \mathrm{MTC_M}$ represents the residual connection and $\mathrm{MTC_M} = \mathrm{MATT} \oplus \mathrm{MATT_F}$.

3.3 Decoding Layer and Classification Layer

BiLSTM or BiGRU is used as the decoding layer to enhance the learning ability of the model. The LMHACL network layer is deep. The contextual relevance and the text sequence information of the text vector are affected after the feature extraction of the above layers. Therefore, we decode the previous layer network. LMHACL can learn the contextual relevance and the text sequence information of the fraudulent phone text again. The vectors from the previous layer pass through the gate structure and the bidirectional network structure again which can further enhance the ability of LMHACL to learn this structure. The calculation formulas are as follows.

$$\overrightarrow{\overline{\mathrm{H_L}}} = \overrightarrow{\mathrm{LSTM}}\,(\mathrm{FFN2} \oplus \mathrm{M})$$

$$\overleftarrow{\overline{\mathrm{H_L}}} = \overleftarrow{\mathrm{LSTM}}\,(\mathrm{FFN2} \oplus \mathrm{M})$$

$$\overrightarrow{\overline{\mathrm{H_G}}} = \overrightarrow{\mathrm{GRU}}\,(\mathrm{FFN2} \oplus \mathrm{M})$$

$$\overleftarrow{\overline{\mathrm{H_G}}} = \overleftarrow{\mathrm{GRU}}\,(\mathrm{FFN2} \oplus \mathrm{M}) \tag{13}$$

$$\overline{\mathrm{H_L}} = \left[\overrightarrow{\overline{\mathrm{H_L}}}, \overleftarrow{\overline{\mathrm{H_L}}}\right]$$

$$\overline{\mathrm{H_G}} = \left[\overrightarrow{\overline{\mathrm{H_G}}}, \overleftarrow{\overline{\mathrm{H_G}}}\right] \tag{14}$$

where $\overrightarrow{\overline{\mathrm{H_L}}}$ and $\overrightarrow{\overline{\mathrm{H_G}}}$ represent the forward hidden layer outputs of LSTM and GRU, respectively. $\overleftarrow{\overline{\mathrm{H_L}}}$ and $\overleftarrow{\overline{\mathrm{H_G}}}$ represent the backward hidden layer outputs of LSTM and GRU. $\overline{\mathrm{H_L}}$ and $\overline{\mathrm{H_G}}$ represent the outputs of the Decoding Layer. $\mathrm{FFN2} \oplus \mathrm{M}$ represents the residual connection, $\mathrm{M} = \mathrm{MTC} \oplus \mathrm{MTC_M}$.

To stabilize the output vector of the hidden layer of the Decoding Layer and enhance the fitting ability of the model, the output vector is normalized in this paper [26]. The calculation formulas are as follows.

$$\mu = \frac{1}{\mathrm{n}} \sum_{\mathrm{i}=1}^{\mathrm{n}} \overline{\mathrm{h_i}} \tag{15}$$

$$\sigma = \sqrt{\frac{1}{\mathrm{n}} \sum_{\mathrm{i}=1}^{\mathrm{n}} \overline{\mathrm{h_i}} - \mu^2} \tag{16}$$

$$\overline{\mathrm{h_i}}^{'} = f\left[\frac{\gamma}{\sigma} \odot \left(\overline{\mathrm{h_i}} - \mu\right) + b\right] \tag{17}$$

where $\overline{\mathrm{h_i}}$ represents the hidden layer output of the decoder, γ is the scaling parameter, b is the bias, $\odot$ means multiplication. $\overline{\mathrm{h_i}}^{'}$ represents the normalized hidden layer output. At this time, $\overline{\mathrm{H}} = \left[\overline{\mathrm{h_1}}, \overline{\mathrm{h_2}}, \ldots, \overline{\mathrm{h_n}}\right]$ is normalized to $\overline{\mathrm{H}}' = \left[\overline{\mathrm{h_1}'}, \overline{\mathrm{h_2}'}, \ldots, \overline{\mathrm{h_n}'}\right]$.

Through the spatial linear transformation of the linear layer, the dimension of the vector is reduced, and the parameters of the model are reduced. The calculation is shown in formula (18).

$$O = W^T \overline{H}' + b_o \tag{18}$$

The sequence structure, contextual relevance, keyword information, global interaction information, and multi-granularity local interaction information of fraud phone text are fused into the output vector.

Softmax is used to classify fraudulent phone text. The calculation is shown in (19).

$$P = \mathrm{SoftMax}\,(O) \tag{19}$$

where $O = \{O_1, O_2, \ldots, O_n\}$ represents the text feature vector and the number of texts is n. The probability value $P = \{P_1, P_2, \ldots, P_n\}$ will be output in this layer and the range is 0–1 which corresponds to each text feature vector respectively. When the possibility is more than 0.5, the model will divide the text into fraud categories.

3.4 Fraudulent Phone Text Recognition Process and Algorithm

Fig. 4 shows the flow chart of fraudulent phone text recognition based on the model LMHACL, which is mainly divided into three parts. The first part is the construction and processing of the data set. The second part is the construction of the classification model. The third part is the performance evaluation and analysis of the model. In the research to realize the recognition of fraudulent phone text, we build the relevant data set first, which includes data cleaning, data annotation, participle, and stopword removal. Next, we build a deep model LMHACL to extract various types of information from the fraudulent phone text and then classify the data. Finally, we analyze the performance of the model. The algorithm for classification is shown below.

```
Algorithm: Fraudulent phone text recognition using the model LMHACL
Input: A labeled fraudulent phone text dataset
A list of the fraudulent phone text, X contains Fraud and Normal sentences. Vectorization representation using Word2Vec.
Where, $X = \{X_n\}_{n=1}^{N}$ is associated with labels $Y = \{Fraud, Normal\}$
Output: Accuracy, Precision, Recall, and F1 of the model LMHACL.
Initialization: All weight parameters and bias parameters of the proposed architecture.
Repeat:
    1. Word vector matrix embedding;
    2. The vector matrix passes through the feature extractor (Embedding Matrix >> BiGRU >> MHAC >> BiGRU). A vector matrix containing multiple fraud semantic information is obtained;
    3. Layer normalization operation to stabilize the vector after feature extraction. According to Eqs. (15)~(17).
    4. Linear layer reduces the dimension of the vector. Category probability is output by Softmax. Data is classified by fraud potential. The Cross-Entropy Loss is calculated. Simultaneously backpropagate the gradient and update the new parameters.
    5. According to Eqs. (20)~(23), calculate Accuracy, Precision, Recall, and F1 of the model LMHACL.
Until: a fixed number of iterations.
Return: Accuracy, Precision, Recall, and F1
```

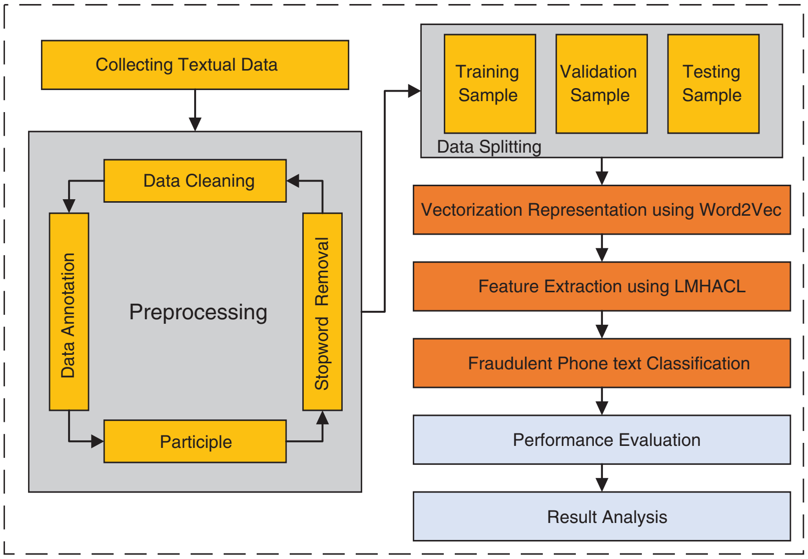


Figure 4: Flow chart of fraudulent phone text recognition based on the model LMHACL

4 Experiment

4.1 Experiment Environment

Windows 10 64-bit operating system, the CPU is Intel(R) Core (TM) i7-10700H CPU @2.90 GHz, the memory capacity is 16 GB, and the GPU is NVIDIA GeForce RTX 2060, the video memory capacity is 6 GB, and the Python version is 3.7. 9, Pytorch is used for the experiments.

4.2 Dataset

Experiment on two datasets.

THUCNews data set: This is a public data set, the content is news headlines, a total of 10 categories, and a total of 200,000 pieces of data, including 180,000 training sets, 10,000 validation sets, and 10,000 test sets. The statistics of THUCNews data are shown in Table 1.

Table 1: THUCNews dataset statistics

Dataset	Training set	Validation set	Test set
Finance	18000	1000	1000
Realty	18000	1000	1000
Stocks	18000	1000	1000
Education	18000	1000	1000
Science	18000	1000	1000
Society	18000	1000	1000
Politics	18000	1000	1000
Sports	18000	1000	1000
Game	18000	1000	1000
Entertainment	18000	1000	1000
Total	180000	10000	10000

Fraud phone text data set: The data is obtained from Baidu, Zhihu, Weibo, Sohu, and other major websites, and some fraud text data sets are manually compiled and modified. We replace all the names in the case. And there is no mention of personal information in the data set. The content includes various types of fraud such as finance, education, postal delivery, banking, making friends, swiping bills, winning lottery tickets, impersonating a police officer, etc., covering almost all types of fraud. There are a total of 10,200 pieces of data, divided into two categories, 5,101 pieces of fraud data, 5,099 pieces of normal data, 6,000 pieces of the training set, 3,000 pieces of the validation set, and 1,200 pieces of the test set. The statistics of fraudulent phone text data are shown in Table 2.

Table 2: Fraudulent phone text dataset statistics

Dataset	Training set	Validation set	Test set
Fraud	3000	1500	601
Normal	3000	1500	599
Total	6000	3000	1200

4.3 Evaluation Method

Evaluation methods: Accuracy, Precision, Recall, and F1-Score. These evaluation methods are calculated as follows:

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} (TP_i + TN_i)}{\sum_{i=1}^{N} (TP_i + TN_i + FP_i + FN_i)} \tag{20}$$

$$\text{Precision} = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N} (TP_i + FP_i)} \tag{21}$$

$$\text{Recall} = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N} (TP_i + FN_i)} \tag{22}$$

$$F1 = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{23}$$

where N represents the number of data categories. TP represents that the real category of the data is positive, and the final prediction result is positive. FP represents that the real category of the data is negative, and the final prediction result is positive. FN represents that the real category of the data is positive, and the final prediction result is negative. TN represents that the real category of the data is negative, and the final prediction result is negative.

4.4 Parametric Experiment

The hyperparameters of LMHACL are determined through a large number of experiments in this paper. AdamOptimizer is chosen as the optimizer, and Word2Vec is used as word embedding with dimension 300. The batch size is 128 and the Dropout is 0.5. In the following experiments on convolution kernels, the product of the convolution kernel and the number of kernels of each scale should be equal to the output size of the BiGRU/BiLSTM hidden layer, so the hidden layer size is selected to be 150. And we choose 0.0001 as the learning rate. The hyperparameters used in this experiment are listed in Table 3.

Table 3: Experiment hyperparameters settings

Hyperparameters	Value or type
Optimizer	AdamOptimizer
Batch size	128
Learning rate	1e-4
Embedding dim	300
Dropout	0.5
Hidden size of BiGRU/BiLSTM	150
Number of head	3
Type of kernel	[1, 2, 3, 4, 5]

We also experimented on the number of heads and type of kernel. The effect of the number of heads on the model performance is investigated by us, 1 to 12 heads are tested in experiments. The experiment is carried out on the fraudulent phone text dataset, and the experiment results of the multi-head attention mechanism are shown in Fig. 5. From the figure, we can see that when the number of heads is 3, it has the highest accuracy. For performance, we chose to use 3 headers.

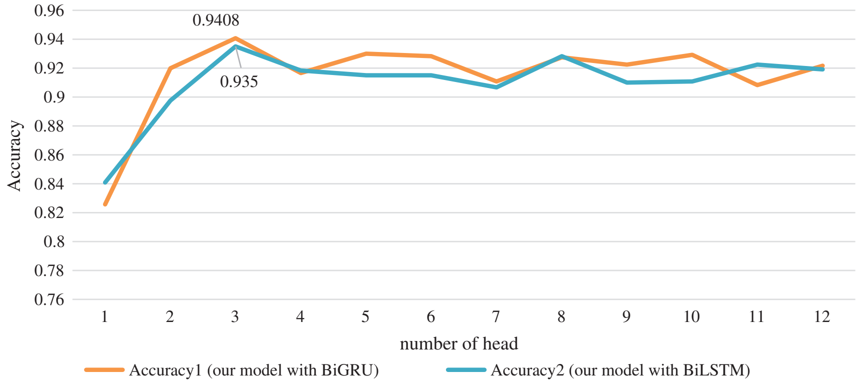


Figure 5: Attention heads number result graph

To obtain the optimal convolution scale of the model, we conducted experiments on the THUCNews dataset and the fraudulent phone text dataset, and the results are shown in Fig. 5. [1, 2, 3, . . . , n] means that kernels with convolution scales of 1, 2, 3, . . . , n are used simultaneously. [1, 2], [1, 2, 3], [1, 2, 3, 4], [1, 2, 3, 4, 5], [1, 2, 3, 4, 5, 6] five various multi-scale convolution kernels are selected for our experiments, and the number of convolution kernels is 150, 100, 75, 60, and 50 corresponding to them, respectively. The product of the convolution kernel of each scale and the number of kernels is equal to the output size of the hidden layer. The detailed data of different scale convolution results are shown in Tables 4 and 5.

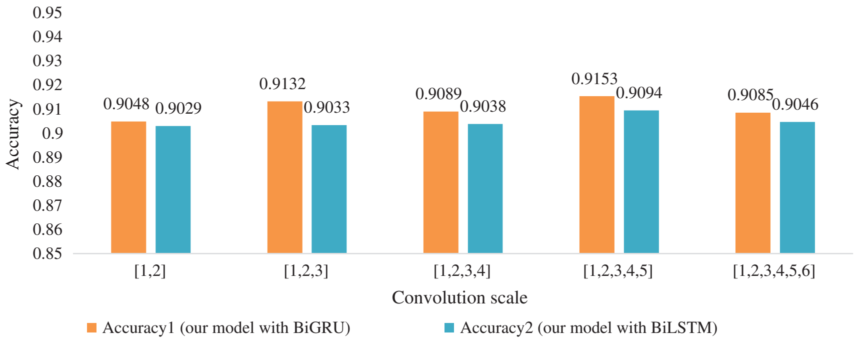
Table 4: Results of the model with different convolution scales on THUCNews

Type of kernel	[1, 2]	[1, 2, 3]	[1, 2, 3, 4]	[1, 2, 3, 4, 5]	[1, 2, 3, 4, 5, 6]
Accuracy (LMHACL with BiGRU)	0.9048	0.9132	0.9089	**0.9153**	0.9085
Accuracy (LMHACL with BiLSTM)	0.9029	0.9033	0.9038	**0.9094**	0.9046
Precision (LMHACL with BiGRU)	0.9044	0.9134	0.9088	**0.9155**	0.9089
Precision (LMHACL with BiLSTM)	0.9035	0.9039	0.9045	**0.9098**	0.9048
Recall (LMHACL with BiGRU)	0.9048	0.9132	0.9089	**0.9153**	0.9085
Recall (LMHACL with BiLSTM)	0.9029	0.9033	0.9038	**0.9094**	0.9046
F1 (LMHACL with BiGRU)	0.9045	0.9130	0.9085	**0.9153**	0.9084
F1 (LMHACL with BiLSTM)	0.9026	0.9033	0.9038	**0.9095**	0.9039

Table 5: Results of the model with different convolution scales on fraudulent phone text

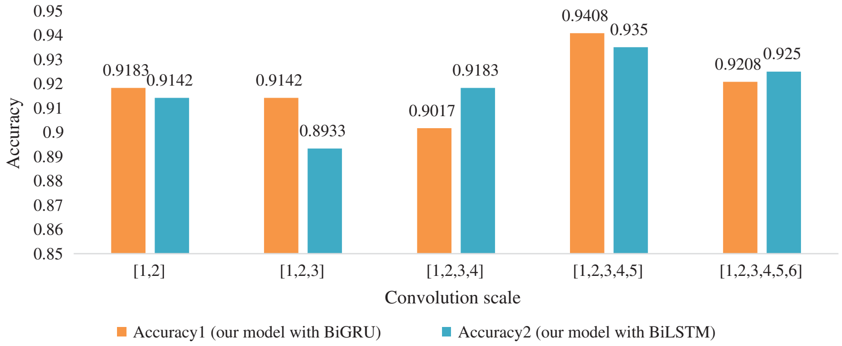
Type of kernel	[1, 2]	[1, 2, 3]	[1, 2, 3, 4]	[1, 2, 3, 4, 5]	[1, 2, 3, 4, 5, 6]
Accuracy (LMHACL with BiGRU)	0.9183	0.9142	0.9017	**0.9408**	0.9208
Accuracy (LMHACL with BiLSTM)	0.9142	0.8933	0.9183	**0.9350**	0.9250
Precision (LMHACL with BiGRU)	0.9220	0.9203	0.9121	**0.9416**	0.9221
Precision (LMHACL with BiLSTM)	0.9207	0.9040	0.9228	**0.9367**	0.9287
Recall (LMHACL with BiGRU)	0.9184	0.9143	0.9018	**0.9409**	0.9209
Recall (LMHACL with BiLSTM)	0.9143	0.8935	0.9184	**0.9351**	0.9251
F1 (LMHACL with BiGRU)	0.9182	0.9139	0.9011	**0.9408**	0.9208
F1 (LMHACL with BiLSTM)	0.9138	0.8926	0.9181	**0.9349**	0.9248

From Fig. 6, Tables 4 and 5, we can see that when the multi-scale convolution kernel of [1, 2, 3, 4, 5] is selected, Accuracy, Precision, Recall, and F1 have the highest values in the experiment. So [1, 2, 3, 4, 5] is selected as the type of kernel in the experiment. Through this convolution kernel, the model learns phrase knowledge of various lengths, and the multi-granularity local interaction improves the classification performance of the model. When the multi-scale convolution kernel of type [1, 2, 3, 4, 5, 6] is selected, the performance index decreases. This phenomenon is caused by the weak semantic connection between words that are too long in a phrase.



(a) THUCNews

Figure 6: (Continued)



(b) Fraudulent Phone Text

Figure 6: Accuracy histogram of different convolution scale model

5 Result and Analysis

The classification performance of the proposed model is compared with several classical text classification models. And an ablation experiment was performed on the normalization layer in the model.

5.1 Classification Methods

This section mainly tells some classification models.

(1) BiGRU [27]: Bidirectional Gated Recurrent Unit, the output of the forward hidden layer and the output of the backward hidden layer are spliced, and the classification result is output by the Softmax.

(2) CNN [28]: Multi-channel TextCNN model, [1, 2, 3, 4, 5] is selected as the size of the convolution kernel, the local interaction features of the text are extracted, and the vectors extracted by each convolution kernel are pooled.

(3) BiLSTM [14]: Bidirectional Long Short-Term Memory Network, the output of the forward hidden layer and output of the backward hidden layer are spliced, and the classification result is output by the Softmax.

(4) BiLSTM-ATT [29]: The same structure as BiLSTM is used, and the attention mechanism is added to highlight the key information of the text.

(5) BiGRU-ATT [30]: The same structure as BiGRU is used, and the attention mechanism is added to highlight the key information of the text.

(6) RCNN [31]: The forward hidden layer output and the backward hidden layer output of the BiLSTM are spliced, and then subjected to maximum pooling. Softmax realizes the classification.

5.2 Model Comparison Using BiLSTM or BiGRU

In LMHACL, BiLSTM or BiGRU is used as the encoder and decoder, respectively, and experiments are performed on the above two datasets. The results are shown in Table 6. In the experiment results, the model using BiGRU is higher than the model using BiLSTM on all evaluation metrics. The accuracy of the model with BiGRU achieved 0.9153 on the THUCNews datasets and 0.9408 on the fraudulent phone text datasets. The model using BiGRU on the THUCNews datasets is 0.0059 more accurate than the model using BiLSTM, and the effect on the fraudulent phone text datasets is

0.0058. The model with BiGRU is better than the model with BiLSTM on the other three benchmarks. Therefore, BiGRU is chosen as the encoding and decoding layers of the LMHACL.

Table 6: Comparing results of the model using BiLSTM and BiGRU

Model	Accuracy	Precision	Recall	F1	
LMHACL with BiLSTM	0.9094	0.9098	0.9094	0.9095	THUCNews
LMHACL with BiGRU	0.9153	0.9155	0.9153	0.9153	
LMHACL with BiLSTM	0.9350	0.9367	0.9351	0.9349	Fraudulent Phone Text
LMHACL with BiGRU	0.9408	0.9416	0.9409	0.9408	

5.3 Ablation Experiment

In this section, the ablation experiments of the normalization layer are performed. Here we show the importance of the normalization layer in our model. The results are shown in Table 7.

Table 7: Results of the ablation experiment

Model	Accuracy	Precision	Recall	F1	
LMHACL with normalization layer	0.9153	0.9155	0.9153	0.9153	THUCNews
LMHACL with no normalization layer	0.9094	0.9100	0.9094	0.9096	
LMHACL with normalization layer	0.9408	0.9416	0.9409	0.9408	Fraudulent Phone Text
LMHACL with no normalization layer	0.9267	0.9289	0.9267	0.9266	

With all other settings unchanged, we conduct experiments with the normalization layer removed on both datasets. The experimental results show that the performance of the model is improved by adding the normalization layer. On THUCNews datasets, the Accuracy, Precision, Recall, and F1 score of the model with the normalization layer are improved by 0.0059, 0.0055, 0.0059, and 0.0057, respectively. On fraudulent phone text datasets, the Accuracy, Precision, Recall, and F1 score of the model with the normalization layer are improved by 0.0141, 0.0127, 0.0142, and 0.0142, respectively. It is proved that the normalization layer can stabilize the output of the model, and the fitting ability of the model is enhanced.

5.4 Experimental Results

Model complexity can be analyzed in terms of time complexity. In general, the function f(n) of module n can represent the number of basic repeated operations of the algorithm. So, the time complexity of the algorithm can be recorded as $T\ (n) = O\ (f\ (n))$. T(n) is proportional to f(n). The smaller f(n) is, the lower the time complexity of the algorithm is, and the higher the efficiency of the algorithm is. O (f(n)) is represented by ε(n) in formula (24), which retains only the highest order and contains no coefficient terms. The time complexity relation is represented in formula (25). To get the order of magnitude ε(n), we ignore the constant and the coefficient of the lower power and the highest power. For example, the time complexity of $f\ (n) = 10n^3 + 5n^2 + 2n + 2$ is $O(n^3)$.

$$If\ \lim_{n\to\infty}\frac{f\ (n)}{\varepsilon\ (n)} \neq 0,\ f\ (n) \sim \varepsilon\ (n) \tag{24}$$

$$O(1) < O(logn) < O(n) < O(nlogn) < O\left(n^2\right) < O\left(n^3\right) < O(2^n) < O(n!) < O(n^n) \quad (25)$$

Although the depth of the model LMHACL is increased, there is only an increase of constant terms in the overall algorithm complexity. In the program, all runnable statements except for loops have a time complexity of O (1). In the program of this experiment, only two layers of loop structure are used at most. The number of loops in the inner loop is 100 and the number of loops in the outer loop is n, so f (n) = 100n. The time complexity is O(n).

We test the efficiency of the model. As shown in Table 8. We evaluate the efficiency of the model by the amount of time consumed. The time consumed by the model during training and testing is recorded. It can be seen that the training time and the testing time of the LMHACL are both maxima. Due to the complexity of fraud semantics, the model LMHACL uses a deeper neural network to extract rich fraud text information, which inevitably reduces the efficiency.

Table 8: Experimental results of the algorithm efficiency comparison

Model	Time 1/s (train)	Time 2/s (test)
CNN	75.8081 s	0.0264 s
BiLSTM	125.7482 s	0.0967 s
BiGRU	107.0258 s	0.0828 s
BiLSTM-ATT	215.1386 s	0.0976 s
BiGRU-ATT	101.6814 s	0.0788 s
RCNN	97.9638 s	0.0983 s
LMHACL	379.6478 s	0.4097 s

The experimental results on the THUCNews dataset and the fraudulent phone text dataset are shown in Tables 9 and 10, respectively. In the above comparison models, the RNN structure with the attention mechanism performs better than the RNN structure without the attention mechanism. In the experimental results of the THUCNews dataset, BiLSTM-ATT is 0.0121 and 0.0115 higher than BiLSTM in accuracy and F1, and BiGRU-ATT is 0.0091 and 0.0088 higher than BiGRU in accuracy and F1, respectively. In the experimental results on the fraudulent phone text dataset, BiLSTM-ATT is 0.0108 and 0.0111 higher than BiLSTM in accuracy and F1, and BiGRU-ATT is 0.0034 and 0.0033 higher than BiGRU in accuracy and F1, respectively. The model proposed in this paper uses a multi-head attention mechanism structure, which can focus on different key information from multiple perspectives. It is better than the general attention mechanism.

The results of LMHACL in this paper are higher than the comparison model on both datasets. The model with the lowest evaluation indicators on the THUCNews dataset is BiLSTM, and the highest is BiGRU-ATT. The accuracy and F1 of the proposed model are 0.0348 and 0.0345 higher than BiLSTM, and 0.0177 and 0.0181 higher than BiGRU-ATT, respectively. The model with the lowest evaluation indicators on the fraudulent phone text dataset is CNN, and the highest is BiLSTM-ATT. The accuracy and F1 of the LMHACL are 0.0416 and 0.0423 higher than CNN, and 0.0275 and 0.0278 higher than BiLSTM-ATT, respectively.

Table 9: Experimental results of each model on the THUCNews dataset

Model	Accuracy	Precision	Recall	F1
CNN	0.8933	0.8943	0.8933	0.8935
BiLSTM	0.8805	0.8819	0.8805	0.8808
BiGRU	0.8885	0.8885	0.8885	0.8884
BiLSTM-ATT	0.8926	0.8925	0.8926	0.8923
BiGRU-ATT	0.8976	0.8973	0.8976	0.8972
RCNN	0.8931	0.8943	0.8931	0.8934
LMHACL	**0.9153**	**0.9155**	**0.9153**	**0.9153**

Table 10: Experimental results of each model on the fraudulent phone text dataset

Model	Accuracy	Precision	Recall	F1
CNN	0.8992	0.9098	0.8993	0.8985
BiLSTM	0.9025	0.9132	0.9026	0.9019
BiGRU	0.9058	0.9126	0.9059	0.9055
BiLSTM-ATT	0.9133	0.9196	0.9134	0.9130
BiGRU-ATT	0.9092	0.9167	0.9093	0.9088
RCNN	0.9017	0.9100	0.9018	0.9012
LMHACL	**0.9408**	**0.9416**	**0.9409**	**0.9408**

The evaluation indicators of LMHACL on the fraudulent phone text dataset are all above 0.94, which reflects the high performance in the recognition of fraudulent phone texts. But the execution time of the model is increased. During the whole training process, the time-series knowledge, context-related knowledge, global interaction knowledge, and multi-granularity local interaction knowledge of fraudulent phone text are all learned by the model, and the output is stabilized by the normalization layer so that fraud semantics can be efficiently identified by the model. Finally, we achieve the efficient classification of fraudulent phone texts.

6 Conclusion

The phone text is classified by the method of text classification to identify fraudulent calls. Compare with general text, fraud semantics are complex and difficult to distinguish. Therefore, we built a fraud phone text classification model and a fraud phone text dataset to allow the model to learn various semantic knowledge of fraudulent phone text. In LMHACL, we use BiGRU as the encoder and decoder and propose a module MHAC to enhance the global and local interaction capabilities of the model. We carry out normalization processing before the classification layer. After experiments, the LMHACL is the best on all the evaluation indexes obtained on the THUCNews and fraudulent phone text datasets compared to other classic models. And the performance indexes on fraudulent phone text data sets are all above 0.94. Thus, it shows the feasibility of the LMHACL in the classification task of fraudulent phone text. However, the efficiency of the model is reduced, and we should reduce the complexity of the model in the future.

Fraud cases occur frequently in society, and the methods of fraud have been changing. In the future, we will continue to pay attention to fraud cases in society and update the data contents. In recent years, large-scale pre-training models have been widely used in the field of text classification. In the future, we will further study the application of Chinese pre-training models in the task of text classification of the fraud phone.

Acknowledgement: The authors wish to express their thanks to one and all who supported them during this work.

Funding Statement: This research was funded by the Major Science and Technology Innovation Project of Shandong Province in China (2019JZZY010120).

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

References

[1] L. Xie, "Research on information investigation of telecom network fraud," *Journal of People's Public Security University of China (Science and Technology)*, vol. 26, no. 3, pp. 85–93, 2020.

[2] H. Z. Ji, Y. C. Ma, S. Li and J. L. Li, "SVM based telecom fraud behavior identification method," *Computer Engineering & Software*, vol. 38, no. 12, pp. 104–109, 2017.

[3] J. Xing, M. Yu, S. P. Wang, Y. R. Zhang and Y. Ding, "Automated fraudulent phone call recognition through deep learning," *Wireless Communications and Mobile Computing*, vol. 2020, no. 2020, pp. 1–9, 2020.

[4] J. J. C. Ying, J. Zhang, C. W. Huang, K. T. Chen and V. S. Tseng, "PFrauDetector: A parallelized graph mining approach for efficient fraudulent phone call detection," in *2016 IEEE 22nd Int. Conf. on Parallel and Distributed Systems (ICPADS)*, Wuhan, China, pp. 1059–1066, 2016.

[5] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu *et al.,* "Deep learning-based text classification: A comprehensive review," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1–40, 2021.

[6] C. Yan, J. Liu, W. Liu and X. Liu, "Research on public opinion sentiment classification based on attention parallel dual-channel deep learning hybrid model," *Engineering Applications of Artificial Intelligence*, Accessed on: September 22, 2022. [Online] Available: https://doi.org/10.1016/j.engappai.2022.105448.

[7] J. K. Tripathy, S. C. Sethuraman, M. V. Cruz, A. Namburu, P. Mangalraj *et al.,* "Comprehensive analysis of embeddings and pre-training in NLP," *Computer Science Review*, vol. 42, no. 1, pp. 1–18, 2021.

[8] V. A. Uymaz and S. K. Metin, "Vector based sentiment and emotion analysis from text: A survey," *Engineering Applications of Artificial Intelligence*, Accessed on: May 13, 2022. [Online] Available: https://doi.org/10.1016/j.engappai.2022.104922.

[9] F. Alotaibi and V. Gupta, "Sentiment analysis system using hybrid word embeddings with convolutional recurrent neural network," *International Arab Journal of Information Technology*, vol. 19, no. 3, pp. 330–335, 2022.

[10] B. A. Chandio, A. S. Imran, M. Bakhtyar, S. M. Daudpota and J. Baber, "Attention-based RU-BiLSTM sentiment analysis model for Roman Urdu," *Applied Sciences*, vol. 12, no. 7, pp. 1–24, 2022.

[11] B. Jang, I. Kim and J. W. Kim, "Word2vec convolutional neural networks for classification of news articles and tweets," *Plos One*, vol. 14, no. 8, pp. 1–20, 2019.

[12] H. Wang and F. Li, "A text classification method based on LSTM and graph attention network," *Connection Science*, vol. 34, no. 1, pp. 2466–2480, 2022.

[13] R. Dey and F. M. Salem, "Gate-variants of gated recurrent unit (GRU) neural networks," in *2017 IEEE 60th Int. Midwest Symp. on Circuits and Systems (MWSCAS)*, Boston, MA, USA, pp. 1597–1600, 2017.

[14] G. Xu, Y. Meng, X. Qiu, Z. Yu and X. Wu, "Sentiment analysis of comment texts based on BiLSTM," *IEEE Access*, vol. 7, no. 2019, pp. 51522–51532, 2019.

[15] X. L. Leng, X. A. Miao and T. Liu, “Using recurrent neural network structure with enhanced multi-head self-attention for sentiment analysis,” *Multimedia Tools and Applications*, vol. 80, no. 8, pp. 12581–12600, 2021.

[16] Y. Yan and K. Zheng, “Text classification model based on multi-level topic feature extraction,” in *2020 IEEE 6th Int. Conf. on Computer and Communications (ICCC)*, Chengdu, China, pp. 1661–1665, 2020.

[17] H. Huan, Z. Guo, T. Cai and Z. He, “A text classification method based on a convolutional and bidirectional long short-term memory model,” *Connection Science*, vol. 34, no. 1, pp. 2108–2124, 2022.

[18] Y. Bao, H. Yang, Z. Yang, S. Liu and Y. Huang, “Text steganalysis with attentional LSTM-CNN,” in *2020 5th Int. Conf. on Computer and Communication Systems (ICCCS)*, Shanghai, China, pp. 138–142, 2020.

[19] J. Liu, Y. Yang, S. L. J. Wang and H. Chen, “Attention-based BiGRU-CNN for Chinese question classification,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 1–12, 2019.

[20] Y. Shi, X. Zhang and N. Yu, “PL-Transformer: A POS-aware and layer ensemble transformer for text classification,” Neural Computing and Applications, Accessed on: October 06, 2022. [Online] Available: https://doi.org/10.1007/s00521-022-07872-4.

[21] M. Tezgider, B. Yildiz and G. Aydin, “Text classification using improved bidirectional transformer,” *Concurrency and Computation: Practice and Experience*, vol. 34, no. 9, pp. 1–12, 2022.

[22] M. A. H. Wadud, M. F. Mridha, J. Shin, K. Nur and A. K. Saha, “Deep-BERT: Transfer learning for classifying multilingual offensive texts on social media,” *Computer Systems Science and Engineering*, vol. 44, no. 2, pp. 1775–1791, 2022.

[23] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun *et al.,* “ERNIE: Enhanced language representation with informative entities,” arXiv preprint arXiv:1905.07129, 2019.

[24] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma *et al.,* “Albert: A lite bert for self-supervised learning of language representations,” arXiv preprint arXiv:1909.11942, 2019.

[25] A. Gulati, J. Qin, C. C. Chiu, N. Parmar, Y. Zhang *et al.,* “Conformer: Convolution-augmented transformer for speech recognition,” arXiv preprint arXiv:2005.08100, 2020.

[26] J. L. Ba, J. R. Kiros and G. E. Hinton, “Layer normalization,” arXiv preprint arXiv:1607.06450, 2016.

[27] Y. Liu, C. Liu, L. Wang, Z. Chen and Y. Wu, “Chinese event subject extraction in the financial field integrated with BIGRU and multi-head attention,” in *2020 Int. Symp. on Automation, Information and Computing (ISAIC)*, Beijing, China, pp. 1–8, 2021.

[28] B. Guo, C. Zhang, J. Liu and X. Ma, “Improving text classification with weighted word embeddings via a multi-channel TextCNN model,” *Neurocomputing*, vol. 363, no. 1, pp. 366–374, 2019.

[29] Q. Jin, X. Xue, W. Peng, W. Cai, Y. Zhang *et al.,* “TBLC-rAttention: A deep neural network model for recognizing the emotional tendency of Chinese medical comment,” *IEEE Access*, vol. 8, no. 2020, pp. 96811–96828, 2020.

[30] W. Wang, Y. Sun, Q. Qi and X. Meng, “Text sentiment classification model based on BiGRU-attention neural network,” *Application Research of Computers*, vol. 36, no. 12, pp. 3558–3564, 2019.

[31] R. Wang, Z. Li, J. Cao, T. Chen and L. Wang, “Convolutional recurrent neural networks for text classification,” in *2019 Int. Joint Conf. on Neural Networks (IJCNN)*, Budapest, Hungary, pp. 1–6, 2019.