Cluster Computing (2020) 23:2901–2913
https://doi.org/10.1007/s10586-020-03055-9

A new method of emotional analysis based on CNN–BiLSTM hybrid neural network

Zi-xian Liu[1,2] · De-gan Zhang[1,2] · Gu-zhao Luo[1,2] · Ming Lian[1,2] · Bing Liu[1,2]

Received: 2 December 2019 / Revised: 2 December 2019 / Accepted: 21 January 2020 / Published online: 3 February 2020

Abstract
The hybrid neural network model proposed in this paper consists of two main parts: extracting local features of text vectors by convolutional neural network, extracting global features related to text context by BiLSTM, and fusing the features extracted by the two complementary models. In this paper, the pre-processed sentences are put into the hybrid neural network for training. The trained hybrid neural network can automatically classify the sentences. When testing the algorithm proposed in this paper, the training corpus is Word2vec. The test results show that the accuracy rate of text categorization reaches 94.2%, and the number of iterations is 10. The results show that the proposed algorithm has high accuracy and good robustness when the sample size is seriously unbalanced.

Keywords Convolutional neural network · Hybrid neural network · BiLSTM · Affective analysis · Text categorization

1 Introduction

In recent years, the mental health of college students has become a social problem which has been paid more and more attention. It is reported that about 20% of college students in China have different degrees of mental problems. Mental problems and mental disorders have become one of the important reasons why freshmen drop out of college every year. This paper mainly uses natural language processing to study the psychological problems of College students, in order to adjust and improve the unreasonable belief system and values of College students. In the research, college students can send their own problems to medical robots in the form of text. Medical robots analyze the emotions of college students by inputting text. Judging adolescents' psychological state according to the existing characteristic parameters, and then finding out the potential psychological problems existing in the students' group, thereby reducing the probability of extreme terrorist events caused by students' psychological problems to a certain extent.

Text affective analysis is an important research branch in the field of natural language processing. Text sentiment analysis, also known as opinion mining, is a process of analyzing and processing emotional subjective texts. At present, the emotional analysis technology for text content mainly focuses on objective information, while the criteria for judging emotional words are determined by people's subjectivity. However, in the process of using medical robots to see a doctor, there will be a large number of subjective texts with emotional significance. It enables researchers to study the emotional analysis of text, from simple analysis of emotional words to more complex emotional sentences or the whole emotional discourse.

De-gan Zhang, Gu-zhao Luo and Ming Lian are co-first author.

✉ Zi-xian Liu
1084256477@qq.com

De-gan Zhang
gandegande@126.com

Gu-zhao Luo
luoguzhao2017@163.com

Ming Lian
limingliangming@gmail.com

Bing Liu
liubing@email.tjut.edu.cn

[1] Key Laboratory of Computer Vision and System, Ministry of Education, Tianjin University of Technology, Tianjin 300384, China

[2] Tianjin Key Lab of Intelligent Computing & Novel software Technology, Tianjin University of Technology, Tianjin 300384, China

Deep learning technology has gradually replaced the traditional machine learning method and become the mainstream text classification technology. Deep learning can express objects more accurately, and it can automatically acquire features of objects from massive data. Learning models based on these functional attributes include convolutional neural network (CNN) [1], recurrent neural network [2], and recurrent neural network [3]. How to effectively classify these massive text data causes expert research. Cui et al. optimized the support vector machine (SVM) algorithm to improve the classification accuracy of text classifier [4]. Yongliang Wu et al. pointed out that traditional machine learning methods are used for classification. TF–IDF model is used to extract category keywords and cosine similarity calculation by which keywords are executed and text keywords to be categorized [5]. Yao et al. [6]. proposed an implicit dirichlet-based text categorization distribution LDA model and SVM algorithm, but to a large extent the number of short text. When the text length is short, the classification effect is not good and the noise is too high. Xia et al. used convolutional neural network to extract features of news text and then realized text classification. Although this method can extract features well, it is easy to ignore the context and make the text semantics inaccurate [7]. Based on the above considerations, this paper uses BI-LSTM-CNN to solve the problem of large-scale enterprise classification—scaling news text. In order to extract feature text better. In this paper, BiLSTM model is used to obtain the representation of two directions, and then the representation of two directions is combined into a new expression through convolution neural network. Each word expression itself is added to the left text vector and the right text vector to be indicated. For left and right text, loop structure is adopted, which is a word on the non-linear transformation and text on the left. This method can better preserve context information and wider word order range [8, 9].

The experiment in this paper uses Word2vec database to train and test, and uses various benchmark models to compare. The experimental results show that the proposed model has better advantages than other models.

The main contributions of this paper are as follows:

(1) Using BiLSTM instead of traditional RNN and LSTM, BiLSTM solves the problem of gradient disappearance or gradient explosion in traditional RNN; at the same time, the semantics of a word is related to the information before and after it, while BiLSTM fully considers the meaning of words in the context and overcomes the drawback that LSTM cannot consider the information after words.
(2) Integrating convolutional neural network and BiLSTM can not only utilize the advantages of convolutional neural network in extracting local features, but also take into account the advantages of bi-directional long-term and short-term memory network in global features of text sequences. BiLSTM is used to solve the problem that the convolutional neural network ignores the context meaning of words in text classification, which improves the accuracy of feature fusion model in text classification.

2 Related works

2.1 Deep learning

In recent years, deep learning algorithms have achieved excellent results in the field of natural language processing. Convolutional Neural Network (CNN) makes full use of the structure of multi-layer perceptrons and has a good ability to learn complex, high-dimensional and non-linear mapping relationships. It has been widely used in image recognition and speech recognition tasks, and achieved good results [10, 11]. Kalchbrenner et al. proposed the application of CNN in natural language processing, and designed a dynamic Convolution Neural Network (DCNN) model to process text of different lengths [12]. The model of English text categorization proposed by Kim takes preprocessed word vectors as input and uses convolutional neural network to achieve sentence-level categorization tasks [13]. Although convolutional neural network has made great breakthroughs in text classification, convolutional neural network pays more attention to local features and ignores the context meaning of words, which has a certain impact on the accuracy of text classification [14, 15]. So this paper uses Bidirectional Long Short-Term Memory (Bi LSTM) network to solve the problem that the convolutional neural network model ignores the context meaning of words.

Neural networks play an increasingly important role in automatic learning and expression of features. For the serialized input, the Recurrent Neural Network (RNN) can integrate the adjacent location information effectively and deal with the tasks of natural language processing. Long Short-Term Memory (LSTM) is a sub-class of RNN [16]. It can be used as a complex nonlinear unit to construct a large-scale neural network structure. It can avoid the gradient disappearance of RNN and has a stronger "memory ability" [17]. It can make good use of the context feature information and the ability to fit the non-linear relationship, and retain the sequential information of the text [18, 19]. RNN has many varieties of cyclic neural network models. Bidirectional RNN is mainly used in text categorization

[20]. Because the semantic information of words in text is not only related to the information before words, but also to the information after words. Bidirectional cyclic neural network which Formed by the combination of two RNNs can further improve the accuracy of text classification.

2.2 Emotional analysis

Emotional analysis is a new research proposition rising in recent years, which has great research and application value. In 2002, Bloomberg put forward the idea of emotional analysis, which attracted wide attention, especially in the final stage of online commentary. Hatzivasiloglou et al. studied the lexical level of emotional orientation [21]. In this paper, they extract adjective relationships from large-scale corpus, and analyze the emotional polarity of these adjectives by logistic regression. Then, adjectives are grouped according to clustering. The accuracy of the results was 82%. Pang et al. used Film commentary as experimental corpus, adopted three machine learning classification methods: native Bayesian, maximum entropy model and support vector machine model, and drew lessons from traditional natural language processing in text categorization technology [22].

Tuney et al. used point-to-point information to determine the emotional polarity of statements, and proposed a method to extract subjective sentences and classify emotions first. In this method, the adjective seed set is used to score the words in the sentence, and then the emotional tendency is judged according to the equivalence [23]. Lin et al. constructed three unsupervised emotional analysis systems using LSM model, JST model and reverse JST model. However, because deep emotional analysis inevitably involves semantics analysis, and often occurs in the text of emotional transfer phenomenon, deep semantic-based emotional analysis method is not ideal [24]. Therefore, in order to improve the effectiveness of deep semantic analysis, a dual LSTM model is introduced in this paper.

3 Construction of model

3.1 Sentence matrix

x_i denotes the word vector corresponding to the ith word in a sentence, each of which has a dimension of 300. Because the number of words in a sentence varies, the sentence is expanded to the same length by adding 0. Then, a sentence with length n can be expressed as [25, 26]:

$$X_{i,n} = x_1 + x_2 + \cdots + x_n \tag{1}$$

Formula (1): " + " denotes the longitudinal connection of word vectors. Then, by using Google's word vector, all sentences can be transformed into sentence matrices $X_{1,n} \in R^{n\times 300}$ in the same size as the input of the model.

3.2 Convolutional neural network

Convolutional neural network (CNN) is an improvement of error back propagation network (BP). Its structure can effectively reduce the computational complexity of traditional BP neural network [27]. The core idea of convolution neural network is: local perception, weight sharing and down sampling. By obtaining some degree of displacement, scale and deformation invariance, the operation speed and accuracy can be improved. In order to use convolution neural network for feature extraction, a deep convolution neural network needs to be trained. Therefore, a simple and effective framework of deep learning Caffe is established [28]. Through the dialogue between users and intelligent medical robots, a large amount of text data information is obtained. When convoluting these data information, we first use pool to reduce the dimension. Then the word $W(i)$ is transformed into the corresponding word vector $\mathrm{V}(W(i))$ by word 2vec, and the sentence composed of the word $W(i)$ is mapped to the sentence matrix S_j. This paper chooses conv3, conv5 and fc7 layers as strong fusion representations. The characteristics of conv1, conv2 and conv4 have less generalization ability and have no significant contribution to the results.

After feature mapping is extracted, the feature is mapped to a fixed size vector. Let the m-layer with time t get the feature mapping to F_m^t, whose vector dimension is $H_f \times W_f \times C \times T$ dimension, H_f is the height of the feature mapping, W_f is the width of the feature map, C is the channel of the feature map, T is the total number of images. The obtained feature mapping set along time domain is as follows:

$$Desc = \sum_{j=1}^{N} F_m^{tj}(x, y) \tag{2}$$

Where J is the number of Jth frames and N is the total number of frames.

In order to overcome the difficulty of feature extraction and classification model transforming into statistical model, a seven-layer convolution neural network structure is proposed. The structure of convolution neural network consists of one input layer (L_0), two convolution layers (L_1 and L_2), two pooling layers (L_3 and L_4), one full connection layer (L_5) and one output layer(L_6) [29]. The structure of convolution neural network is shown in Fig. 1.

L_0: 20 filters in convolution layer, each convolution layer filter dimension is 5 × 5; L_2: The area size of maximum pooling layer is 10 × 10; L_3: 30 filters in convolution layer, each convolution layer filter dimension is 5 × 5;

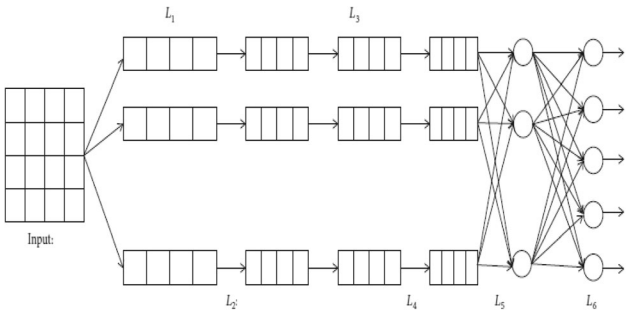

Fig. 1 Convolutional neural network structure

L_4: The area size of maximum pooling layer is 10 × 10; L_5: 500 units in full connection layer; L_6: Softmax classifier with 5 units in output layer.

The details of each layer in the proposed CNN architecture are shown in Table 1.

CNN starts with an input matrix, followed by a shaping operation to convert data into a specific format. Then, two convolution blocks and the largest pool block are applied continuously. Each block convolutes its input signal with an estimated kernels of step size 1 to 5, and step 1 is designed to extract a specific number of feature maps from the data. Then, the maximum pool layer (pool size n = 10) is used to reduce output. This layer is designed to reduce the size of the feature graph while keeping the number unchanged. Then, the output of two pooling layers is put into a full connection layer, and the output features of the last layer are aggregated by the full connection layer to form a global feature for text emotional classification. Finally, these features will be input into the last layer with five neurons, and the probability vectors will be obtained by using soft max.

3.3 Bidirectional long-term and short-term memory

For the acquired text information, the output of emotional classification depends on the current input and previous state. Assuming that a given input sequence is represented by $x = \{x_1, x_2, \ldots, x_t, \ldots, x_T\}$, where t represents frame t and the total number of frames is frame T, the following formula is obtained [30, 31]:

$$h_t = \sigma_h(W_{xh}x_t + W_{hh}x_{t-1} + b_h) \tag{3}$$

Whereas h_r represents the output of the hidden layer at t time, W_{xh} represents the corresponding weight matrix from the input layer to the hidden layer, W_{hh} is the weight matrix from the hidden layer to the hidden layer, b_h is the deviation of the hidden layer, and σ_h is the activation function. Finally, the output can be obtained by the following equation:

$$y_t = \sigma_y(W_{ho}h_t + b_o) \tag{4}$$

Where y_t represents the predicted value of the t-th sequence, W_{ho} represents the weight matrix from the hidden layer to the output layer, b_o is the output deviation, σ_y represents the activation function.

The main problem of RNN is that it can only model short time series, because the error gradient will disappear rapidly as the network changes deeply. To solve this problem, LSTM introduces three gates to maintain state. As shown in Fig. 2, there are three gates, including input gate (i_t), forget gate (f_t) and output gate (o_t).

If it controls the flow of information into or out of the network, f_t controls the impact of previous sequences. The solutions are as follows:

$$\begin{cases} i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \\ f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \\ o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \\ c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ h_t = o_t \odot \tan c_t \end{cases} \tag{5}$$

Among them: c_t represents the storage unit at time t, h_t represents the output of hidden layer, b_α represents the deviation, where $\alpha \in \{i, f, c, o\}$, weighted parameters

$$W = \left\{W_{xi};\ W_{xo};\ W_{xf};\ W_{ci};\ W_{co};\ W_{cf};\ W_{hi};\ W_{ho};\ W_{hf}\right\}$$

obtained by time back propagation [32].

Table 1 Parameters of the CNN architecture

Layer	Type of layer	Number of units	Convolution or pooling kernel size	Step size	Output dimension
L_0	Input layer	6000			(N,T,1)
L_1	Convolution layer	20	1 × 5	1 × 1	(N,T,20)
L_2	Maximum pooling layer		1 × 10	1 × 10	(N,T,20)
L_3	Convolution layer	30	1 × 5	1 × 1	(N,T,600)
L_4	Maximum pooling layer		1 × 10	1 × 10	(N,T,600)
L_5	Full connection layer	500			(N,T,500)
L_6	Softmax	5			(N,T,5)

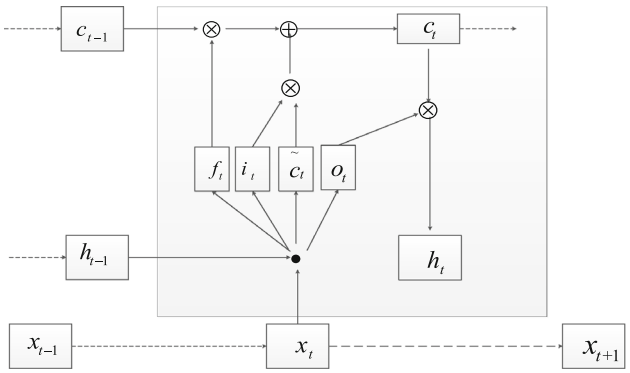

Fig. 2 LSTM memory module structure

3.4 Model training

Although LSTM can capture long text information, it only considers one direction. In other words, LSTM assumes that the current text is only affected by the previous text frame, but the following frame data is also related to the current state. We hope to strengthen this two-way relationship. This means that when dealing with the current text frame, we also need to consider the next text frame. Bi-LSTM is very suitable for this problem. Our Bi-LSTM model is shown in Fig. 4. The first layer is forward LSTM, and the second layer is backward LSTM.

Bi-LSTM can solve the relationship between the two text frames and strengthen the bidirectional relationship between the current text frame and the next text frame. This is because Bi-LSTM can model the bidirectional time structure, so it can capture more structural information and perform better than one-way LSTM [33]. Thus, the CNN–BiLSTM model can be obtained as shown in Fig. 3.

From the figure above, we can see that the features are acquired through CNN, and then the output of CNN is passed through LSTM in text time order. LSTM links the output of the underlying CNN as the input of the next moment [34].

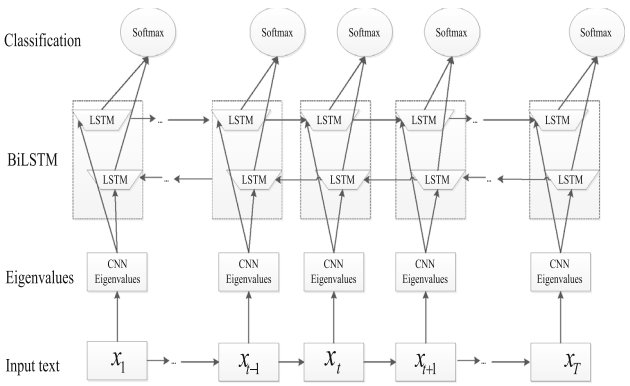

Fig. 3 CNN–BiLSTM

The first layer is forward LSTM, and the second layer is backward LSTM. The final output can be calculated by the following formula:

$$\begin{aligned} h_t &= \alpha h_t^f + \beta h_t^b \\ y_t &= \sigma(h_t) \end{aligned} \tag{6}$$

Where h_t^f represents the output of forward LSTM layer, it takes the sequence from x_1 to x_T as input, h_t^b represents the output of backward LSTM. The sequence from x_1 to x_T, α and β controls the factors ($\alpha + \beta = 1$) of forward LSTM and backward LSTM. h_t represents the sum of two unidirectional LSTM elements at time t. σ here is the softmax function, y_t is the predicted value.

4 Design of CNN–BiLSTM algorithm

4.1 Idea of CNN–BiLSTM algorithm

4.1.1 Related definitions

Definition 1 For a hybrid neural network with M inputs $X \in R^M$, the output is Y, the connection weight is W, and the node function is in the form of hard limit. There is the following equation:

$$Y = \mathrm{sgn}(W \circ X) = \begin{cases} 1, & \bigvee_{m=1}^{M} (w_m \wedge x_m) \geq 0 \\ -1, & \bigvee_{m=1}^{M} (w_m \wedge x_m) < 0 \end{cases} \tag{7}$$

Let D be the expected output of the sample, and the weight modification rule is: set the initial weight $W^0 \neq 0$,

$$\text{If } W^k \circ X < 0 \text{ and, then } W^{k+1} = W^k + X \tag{8}$$

$$\text{If } W^k \circ X > 0 \text{ and } D < 0 \text{ then } W^{k+1} = W^k - X \text{ else } W^{k+1} = W^k \tag{9}$$

If W and t make $f = \mathrm{sgn}(W \circ X - t)$, then function $f : R^n$ (or R^n subset) $\rightarrow \{-1, +1\}$ is linear separable.

The convergence theorem of hybrid neural networks expressed in Ref. [35] is: if the learning function is linear separable and the sample size satisfies $\bigvee_{m=1}^{M} x_m = 1, x_m \in [0, 1], (m = 1, 2, \ldots, M)$, after preprocessing, the hybrid neural network perceptron can converge to the correct value through the learning process (7)-(9) after finite iterations.

Theorem 1: *If the learning function is linearly separable, and the sample size satisfies* $\bigvee_{m=1}^{M} x_m = 1, x_m \in [-1, 1](m = 1, 2, \ldots M)$, *after*

pretreatment, and $p \in [1, M]$ exists, so that $x_p < 0$ then the hybrid neural network perceptron can converge to the correct value through the learning process Eqs. (7–9) after a finite number of iterations.

Proof simplify the learning process as follows:

(1) make $k = 1$, initialize $W^k \neq 0$;
(2) appoint $i \in \{1, 2, \ldots, M\}, x^k = (x_{i1}, x_{i2}, \ldots, x_{iM})$;
(1) if $W^k \circ X^k \geq 0$, return (2), otherwise execute (4);
(4) $W^{k+1} = W^k + X^k, k = k + 1$, return (2).

When $Y < 0$, $-X^k = (-x_{i1}, -x_{i2}, \ldots - x_{iM})$ is used instead of $x^k = (x_{i1}, x_{i2}, \ldots, x_{iM})$, then:

$$Y < 0 \Leftrightarrow \bigvee_{m=1}^{M} (w_m \wedge x_m) < 0 \Leftrightarrow w_m \wedge x_m < 0, (m = 1, 2, \ldots, M) \Rightarrow w_p \wedge x_p < 0$$

If $w_p \geq 0$,then:

$w_p \wedge (-x_p) \geq 0 \Rightarrow \bigvee_{m=1}^{M} (w_m \wedge (-x_m) \geq 0 \Leftrightarrow Y > 0.$ If $w_p < 0$, Because $-x_m > 0$, Then it is known through the iteration process (2)–(4). After finite iterations, $w^{k+1} = w^k + (-x_p)$ must have $w_p \geq 0$. Therefore, the above hypothesis is valid and the theorem is proved.

4.1.2 Feature extraction strategy based on CNN

In this paper, convolutional neural network model is used to extract local features. When using convolutional neural network to classify text, the word $W(i)$ is first transformed into the corresponding word vector $V(W(i))$ by word 2vec, and the sentence composed of the word $W(i)$ is mapped to the sentence matrix S_j. $V(W(i)) \in R^k$ represents the i-th word vector in the sentence matrix S_j as the K-dimension word vector. $S_j \in R^k$ represents the number of sentences in the sentence matrix S_j. Sentence matrix S_j is the vector matrix of the embedded layer of the convolutional neural network language model, Where the sentence matrix is expressed as$S_j = \{V(W(1)), V(W(2)), \ldots, V(W(m))\}$.

The convolution layer convolutes the sentence matrix S_j with a filter of size $r \times k$, and extracts the local features of S_j.

$$c_j = f(F \cdot V(W(i : i + r - 1)) + b) \tag{10}$$

Among them, F represents the dimension of the filter r, b represents offset, f represents function of non-linear operation through RELU. $V(W(i : i + r - 1)$ represents r-row vectors from i to $i + r - 1$; c_j represents the local feature obtained by convolution operation. As the filter slides from top to bottom with the step size of 1, and passes through the whole S_j, the local eigenvector set C is finally obtained.

$$C = \{c_1, c_2, \ldots, c_{r-h+1}\} \tag{11}$$

For the local feature obtained by convolution operation, the maximum pooling method is used to extract the feature with the largest value instead of the whole local feature. The pooling operation can greatly reduce the size of the feature vector.

$$d_i = \max C \tag{12}$$

Finally, all the pooled features are combined in the full connection layer, and the output vector U:

$$U = \{d_1, d_2, \ldots, d_n\} \tag{13}$$

Finally, we classify the U output from the full connection layer into the soft Max classifier. The model uses the labels in the actual classifier to optimize the parameters through back propagation algorithm.

$$P(y|U, W, b) = soft\max(F \cdot U + b) \tag{14}$$

4.1.3 BiLSTM network for emotional classification

Although LSTM solves the problem that RNN will undergo gradient disappearance or gradient explosion, LSTM can only learn the information before the current word, but cannot use the information after the current word. Because the semantics of a word is not only related to the previous historical information, but also to the information after the current word, this paper uses Bi LSTM instead of LSTM, which not only solves the problem of gradient disappearance or gradient explosion, but also fully considers the current context information.

Using BiLSTM to learn the sentence matrix $S_j = \{V(W(1)), V(W(2)), \ldots, V(W(m))\}$, the text features obtained are global, and the context information of words in the text is fully considered. The process of global feature extraction and classification using BiLSTM is shown in the following figure.

4.2 Description of the new algorithm

The implementation of CNN–BiLSTM hybrid neural network algorithm is mainly divided into three parts: word vectorization, feature extraction by CNN–BiLSTM hybrid neural network and classifier classification.

(1) Word vectorization: using Word Embedding language model to obtain. Get the input sequence $X = \{x_1, x_2, x_3, \ldots, x_i\}$, which x_i is the input vector of K dimension (in this paper K takes 300) dimension. Then the total input vector X' is obtained by accumulating the average. In this way, as the input of CNN–BiLSTM feature extraction model, the word vector can not only prevent the difficulty of model

training caused by too high dimension, but also obtain more semantic information.

PV_DM (Distributed Memory Model of Paragraph Vectors) is a word vectorization method proposed by Mikolov et al. based on Word2Vec principle for better training of word vectors. This paper uses this method to train word vectors. PV can refer to indefinite text such as phrases, sentences, paragraphs or large documents. In this paper, its representative sentence vector. Compared with CBOW, the structure of PV_DM only has one more text vector to represent the input part of the model. Word vector matrix W is the mapping of words. Word vector of each word is represented as one column. In order to store the current context information, all text vectors identified by unique ID vector are arranged in column in matrix D. All texts share the word vector matrix W. Different texts have different text identifiers, and the same text stored in the context will share the same text identifier. The input of the Softmax layer is the new vector of word vector and text vector through connection or accumulation. Finally, the most probable words are predicted by establishing Huffman tree, in which the leaf node is the word in the text and the weight is the number of words. Maximizing average likelihood estimation is actually the training process of word vectors:

$$P = \frac{1}{T}\sum_{t=k}^{T-k} \log p(w_t|w_{t-k}, \ldots, w_{t+k}) \tag{15}$$

$$p(w_t|w_{t-k}, \ldots, w_{t+k}) = \frac{e^{y_{w_i}}}{\sum_i e^{y_i}} \tag{16}$$

$$y = Uh(w_{t-k}, \ldots, w_{t+k}; W, D) + b \tag{17}$$

Among them, b is the offset and the weight matrix is U; the non-normalized logarithmic probability of each output word is y_i; $h(\cdot)$ is the connection operation, and the Softmax classifier is $p(\cdot)$.

The PV_DM model can be implemented by Doc2vec of Python's gensim library. The process of word vectorization is shown in the following Table 2.

(2) Feature selection and extraction: as shown in Fig. 4, the feature fusion model in this paper consists of convolutional neural network and bidirectional long-short memory network (BiLSTM). The first layer of the convolution neural network is the word embedding layer, which takes the sentence matrix of the word embedding layer as input, the column of the matrix is the dimension of the word vector and the sequence_length of the matrix's behavior; the second layer is the convolution layer, which carries out convolution operation and extracts local features. The text classification parameters of the benchmark convolution neural network are given in Ref. [36]. The analysis shows that when the word vector is 100 dimension, the filter is 3×100, 4×100 and 5×100, which can achieve better classification results. Therefore, 128 filters of 3×100, 4×100 and 5×100 size are selected in this paper. The step size of stride is set to 1, and the padding is VALID. The convolution operation is carried out by convolution operation. To extract the local features of sentences; the third level carries out maximum pooling operation, extracts key features, discards redundant features, generates feature vectors with fixed dimensions, and splices the output features of the three pooling operations as part of the input features of the first level full connection layer.

The first layer of BiLSTM is the word embedding layer, in which the sentence matrix of the embedding layer is used as input, and the dimension of each word vector is set to 100 dimensions. The second and third layers are hidden layers with the size of 128 hidden layers. The current input is related to the sequence before and after, and the input sequence is input from two directions to the model through the hidden layer. The historical and future information of the two directions are saved. Finally, the output parts of the two hidden layers are joined together to get the final output of BiLSTM.

BiLSTM model is used to extract the context semantic information of words and global features of words in text. Before the first Fully Connected Layers (FC), this paper uses concat() method in TensorFlow framework to fuse the features of CNN and BiLSTM output. The fused feature is saved in output, as the input of the first full connection layer, dropout mechanism is introduced between the first full connection layer and the second full connection layer, and some trained parameters are discarded in each iteration, so that weight updating does not depend on some inherent features and avoids over-fitting.

3) Classifier: finally, input to the softmax classifier and output the classification results. In this paper, the probability of classifying x into class j in the soft Max regression is as follows:

$$P(y^{(i)} = j|x^{(i)}; \theta) = \frac{\exp(\theta_j^T x^{(i)})}{\sum_{l=1}^{k} \exp(\theta_l^T x^{(i)})} \tag{18}$$

The essence of text emotional analysis belongs to the category of text categorization, and the ultimate goal of this chapter's training model is to correctly predict whether the emotional categories of the input sentences are positive or negative. The algorithm design of sentiment analysis for

Table 2 Algorithm of word vectorization

Input: large scale unmarked text $T_{unlabelled}$
Output: a set of word vectors for all words

1: Loading corpus $T_{unlabelled}$;
2: Data preprocessing in $T_{unlabelled}$;
3: Initialize PV_DM model parameters gensim.models.doc2vec(dm = 1, size = 100);
4: For each sentencet $\in T_{unlabelled}$
5: For each word $w_t \in t$
6: Get the context words of the target words and calculate $\log p(w_t \in t|w_{t-k}, \ldots, w_{t+k})$;
7: End for
8: Maximizing objective function $\frac{1}{n}\sum_{t=k}^{T-k} \log p(w_t \in t|w_{t-k}, \ldots, w_{t+k})$;
9: End for
10: The set of word vectors of all generated words is derived;

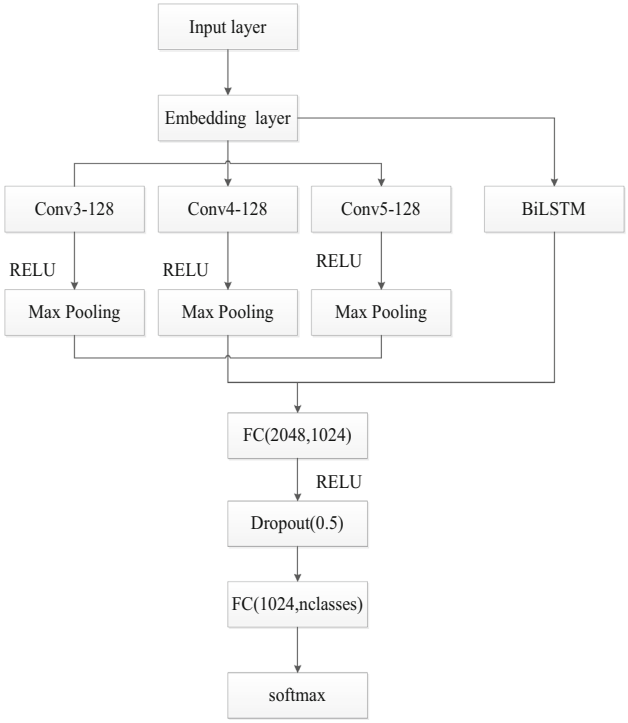

Fig. 4 Feature fusion model of CNN and BiLSTM

commentary text based on CNN–BiLSTM model is presented in this paper as shown in Table 3.

4.3 Algorithmic complexity analysis

4.3.1 Time complexity analysis

The time complexity determines the training/prediction time of the model. If the complexity is too high, the training and prediction of model will take a lot of time, which can neither quickly verify ideas and improve models, nor fast prediction.

Time complexity of a single CNN model is $Time \sim O(M^2 \times K^2 \times Cin \times Cout)$.

M is the size of the output characteristic graph, K is the size of the convolution core (Kernel), Cin is the number of input channels, $Cout$ is the number of output channels, M/K is the number of filters.

Time complexity of BiLSTM model is $Time \sim O(M^2 \times K^2 \times 2Cin \times 2Cout)$.

In order to use CNN–BiLSTM algorithm to compute local features, in the process of calculating local features, a filter of $r \times k$ is required to convolute a pair of sentence matrices with one step size. It needs to scan r-row vectors to get the set of local eigenvectors, and then find out the largest feature instead of the whole local feature. Therefore, it needs r-1. Secondly, the time complexity is $O(r(r-1))$.

Then, we use BILSTM to learn the global features of m sentence matrices. m sentence features are transmitted to BILSTM model through input gates, and the time complexity is $O(Cin)$. It can be seen that the time complexity is greatly reduced by using CNN–BiLSTM algorithm.

4.3.2 Spatial complexity analysis

Spatial complexity determines the number of parameters of the model. Due to the limitation of dimension disaster, the more parameters of the model, the more data needed for training model, and the data set in real life is usually not too large, which will make the training of the model easier to over-fit.

When using CNN–BiLSTM for sentiment analysis, according to formula (7)–(10), we can see that using CNN to extract the local features of text can greatly reduce the size of feature vectors, discard redundant features, and input their features into BILSTM model to extract global features, in the first full connection layer and the second link layer. The dropout mechanism is introduced between the full connection layers, and the trained parameters are

Table 3 Algorithm of the CNN–BiLSTM

Input: corpus T_{train},T_{test} of data
Output: Text Emotional Category Label for Test Data

1: Loading corpus;
2: Data preprocessing of corpus T_{train}, T_{test};
3: Get the set W of word vectors of text;
4: Initialize CNN–BiLSTM model parameters;
5: For each sentence $t \in T_{train}$;
6: The set of word vectors of all the words found forms $t = [w_1, w_2, \ldots, w_{n-1}, w_n]$
7: Obtaining eigenvalue $c_j = f(w_j * m + b)$ by convolution operation
8: Generate expression sequence and output eigenvector $h = [h_1, h_2, \ldots, h_{n-1}, h_n]$ through BLSTM memory storage unit
9: All eigenvalues are combined into eigenvectors $c = [(c_1, h_1), (c_2, h_2), \ldots, (c_n, h_n)]$
10: The most important features $c_j^k = \max(c_{i,j})$ are obtained by maximum pooling method.
11: Back propagation algorithm is used to adjust model parameters and text word vectors;
12: Calculating the probability value

$$P(y^{(i)} = j | x^{(i)}; \theta) = \frac{\exp(\theta_j^T x^{(i)})}{\sum_{l=1}^{k} \exp(\theta_l^T x^{(i)})}$$

of text sentiment label of input samples by using Softmax operation;
13: End for
14: For each sentence $t \in T_{test}$
15: Classification of samples using trained CNN–BiLSTM model
16: Output emotional category label;
17: End for

discarded in each iteration, so that the weight updating is no longer dependent on the inherent characteristics of the part. Thus the number of parameters obtained is less than that of the traditional BILSTM parameter $(n + 1) \times m$(n is the dimension of the input, m is the dimension of the output), which effectively prevents over-fitting. It can be seen from this that the CNN–BiLSTM algorithm is in good agreement with the traditional BILSTM parameter t. Compared with single CNN and BILSTM, the spatial complexity is reduced.

5 Experimental tests

5.1 Experimental testing

In order to verify the effectiveness of the proposed algorithm based on CNN–BiLSTM, TensorFlow is used as the experimental tool. TensorFlow uses data flow diagram to plan the computing process. It can map the computing to different hardware and operating system platforms. In this paper, we use TensorFlow tool and Word2vcc to generate word vectors and implement the training of CNN–BiLSTM model.

5.2 Experimental data

This paper evaluates the emotional classification model using the large data set of movie reviews collected from IMDB. The data set consists of a training set and a test set. It contains 12,500 positive emotional movie reviews and 12,500 negative emotional movie reviews, totaling 50,000 movie reviews. Each comment is composed of a commentary text and a comment group, with a score of 10.

Wikipedia and Reuters RCV1 datasets are selected as corpus. Firstly, Jieba word segmentation tool is used to segment comments, and stop-word documents are used to delete useless words, invalid symbols and punctuation symbols in comments. Then the Word2vec method is used to train the word vectors on the corpus and generate 100-dimensional vectors for each word in the corpus. Word2vec tool is used to configure: words appear at least five times in the corpus; context window size is set to 10.

5.3 Adjustment of parameters

In training model, parameters are adjusted by adjusting the dimension of word vector, word frequency threshold and window size. The dimension of the word vector is tested from 50 to 200. It is found that when the dimension of the word vector is about 120, the F value of the test data is the best, as shown in Fig. 5.

Since the word frequency threshold can not generate the word vector when it is less than 5, it also can not generate the index, so the accuracy of selecting the word frequency threshold to be 5 window size is the highest when it is close to 20 in the training process, as shown in Fig. 6.

When training CNN–BiLSTM, the number of iterations is observed by the loss value. During the experiment, it was found that the loss value remained unchanged after 5 iterations in the BILSTM model, unchanged after 3 iterations in the CNN model, and unchanged at 10 iterations in the CNN–BiLSTM model, as shown in Fig. 7.

5.4 Analysis of experimental results

In order to verify the classification performance of the proposed CNN and BiLSTM feature fusion models, the comparison experiments of the proposed feature fusion model with single CNN model and single BiLSTM model are carried out.

During the experiment, the dimension of word vector is 120, the size of sliding window is 5, the number of sliding windows is 128, the activation function is RELU, the Pooling method is Max, dropout rate is 0.5 and Epoch is 60; the dimension of word vector is 120, the number of layers is 2, the optimization function is Adam, and the learning rate is 0.001. Epoch is 60, the size of hidden layer is 128, and the learning rate is set to 0.001. The accuracy and loss function of CNN model, Bi LSTM model and this model are shown in Figs. 8 and 9.

From the comparison of Fig. 8, it is found that the convergence speed of the fusion model on the test set is

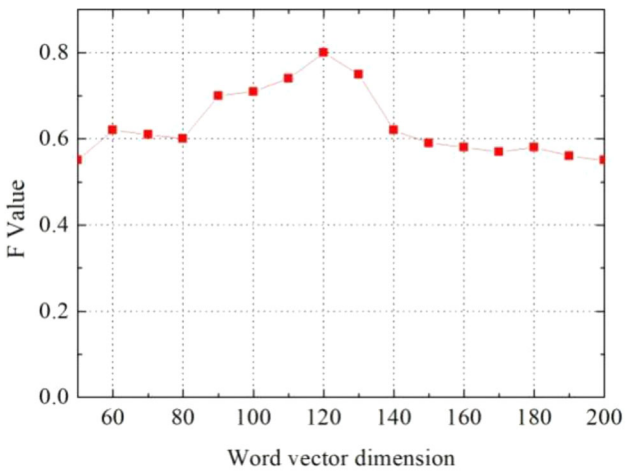

Fig. 5 Change between word vector dimension and F value

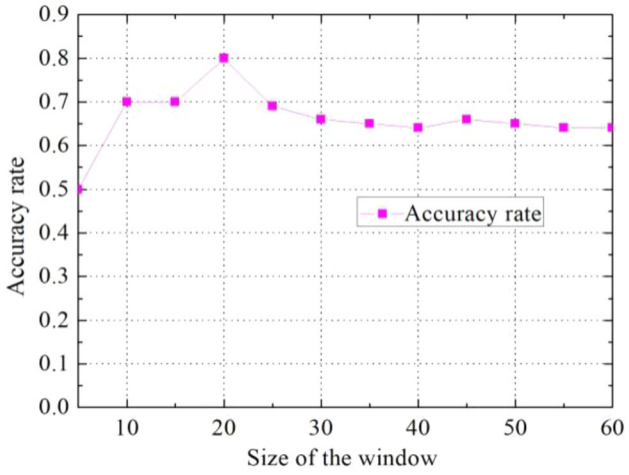

Fig. 6 Changes between window size and accuracy

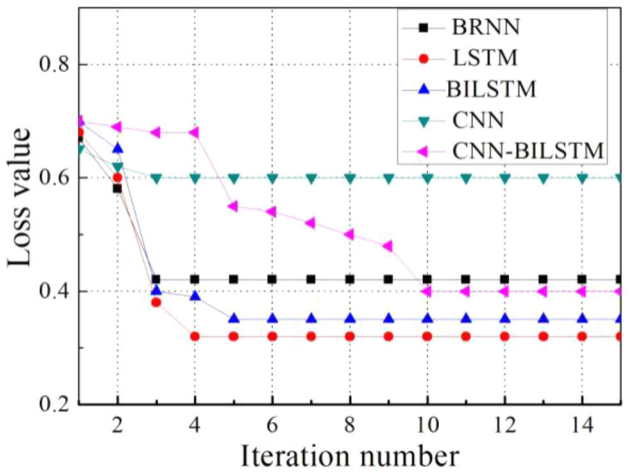

Fig. 7 The change between iteration number and loss value

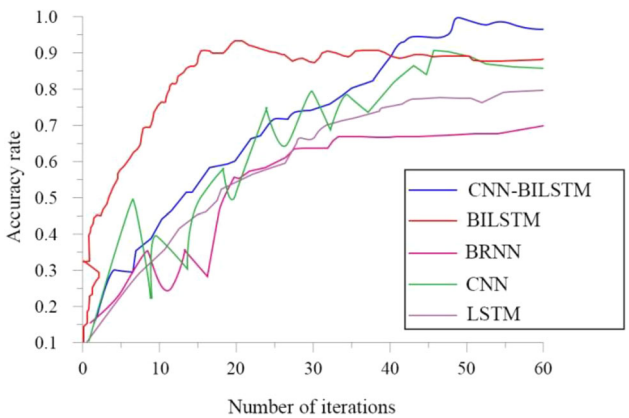

Fig. 8 Accuracy comparison of three models

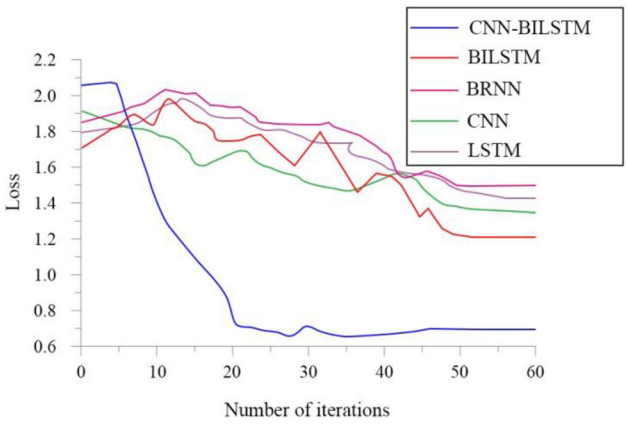

Fig. 9 Loss comparison of three models

slow, but the accuracy of the fusion model is higher than that of the single CNN and Bi LSTM models. Compared with Fig. 9, it is found that the loss value of single CNN and single Bi LSTM model decreases to stable value faster than that of fusion model, but ultimately the loss value decreases to a very low stable value, and the model achieves better convergence effect.

In order to fully illustrate the validity of the model, the precision, recall and F-measure of NLPCC evaluation rules are used as the evaluation indicators of relevant experiments on data sets.

(1) Precision: Precision is a measure of accuracy, representing the proportion of instances that are classified as positive instances that are actually classified as positive data instances.

$$P = \frac{TP}{TP + FP} \tag{19}$$

(2) Recall: The recall rate is a measure of coverage, indicating the proportion of instances correctly classified as positive instances in actual positive emotional data instances.

$$R = \frac{TP}{TP + FN} \tag{20}$$

(3) F-measure: The larger the value, the better the performance of text emotional classification.

$$F = \frac{2PR}{P + R} \tag{21}$$

In the formulas above, TP denotes the number of samples that are actually positive and are correctly classified as positive, FN denotes the number of samples that are actually negative and are correctly classified as negative, FP denotes the number of samples that are actually positive but are incorrectly classified as negative, and the positive examples here not only denote the number of samples that are emotionally inclined to be positive, but also can be used. The negative sample refers to emotional inclination, which represents a certain category, not a certain category.

Table 4 Comparison of single model and fusion model

Model	P (%)	R (%)	F (%)	Accuracy (%)
BRNN	87.3	87.5	88.0	87.3
LISTM	87.9	87.3	88.2	87.5
BILISTM	88.5	87.8	88.4	87.9
CNN	88.1	87.6	88.1	88.0
CNN–BiLISTM	94.3	94.6	94.5	94.2

The CNN–BiLSTM model and BRNN, LISTM, CNN [37], BILISTM [38] and CNN–BiLISTM proposed in this paper are compared. The final results are shown in Table 4.

As can be seen from Table 4, the CNN–BiLISTM model proposed in this chapter has the best effect, and the final accuracy rate reaches 94.2%.

Compared with the traditional BRNN model and LISTM model, the accuracy and F value of BLSTM are improved, reflecting the importance of long-distance dependence on information for text sentiment analysis. Because of the advantages of CNN model in dealing with local features, it can also be seen that the accuracy of CNN–BiLISTM is higher than that of single BLS TM model and single CNN.

6 Conclusions

We propose a CNN architecture which combines two layers of long-term and short-term memory network in this paper. We use convolutional neural network to extract features. Then we input the captured word vectors into Bi-LSTM, model them in two directions, and classify them by using Softmax layer. Finally, the output of three-channel Softmax layer is fused averagely. This model can extract the local features of text effectively by using convolutional neural network, and also take into account the global features of text by using BiLSTM, taking full account of the context semantic information of words. Experiments show that when word vectors constructed by Word2vec model pass through CNN–BiLSTM model, it is helpful to extract the implicit features of word vectors. The fusion model proposed in this paper is compared with single CNN model and single BiLSTM model. The classification accuracy of the fusion model proposed in this paper is better than that of single CNN and single BiLSTM model. The results show that the proposed feature fusion model is superior to the contrast model in classification accuracy. The fusion model in this paper effectively improves the accuracy of text classification.

Funding This research work is supported by Intelligent Psychological Consultation System Based on NLP and User Portraits (201910060022), College Students' Innovative Entrepreneurial Training Plan Program (National Program, 2019.6-2020.6), National Natural Science Foundation of China (Grant No. 61571328), Tianjin Key Natural Science Foundation (No. 13JCZDJC34600), CSC Foundation (No. 201308120010), Major projects of science and technology in Tianjin (No. 15ZXDSGX 00050), Training plan of Tianjin University Innovation Team (No. TD12-5016, No.TD13-5025), Major projects of science and technology for their services in Tianjin (Nos. 16ZXFWGX00010, 17YFZC GX00360), the Key Subject Foundation of Tianjin (15JCYB JC46500), Training plan of Tianjin 131 Innovation Talent Team (No. TD2015-23).

Compliance with ethical standards

Conflict of interest Author Liu Zi-xian, Zhang De-gan, Luo Gu-zhao, Lian Ming and Liu Bing declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Tang, D., Qin, B., Liu, T.: Deep learning for sentiment analysis: successful approaches and future challenges. Wiley Interdiscip Rev **5**(6), 292–303 (2015)
2. Lan, W.F., Xu, W., Wang, T.: Text classification of Chinese news based on convolutional neural network. J. South-Central Univ. Natl. (Nat Sci Edition) **1**, 138–143 (2018)
3. Gong, Q.J.: A text classification based on the Recurrent Neural Networks. Huazhong University, Wuhan (2016)
4. Cui, J.M., Liu, J., Liao, Z.Y.: Research of text categorization based on support vector mechine. Comput. Simul. **30**(2), 299–302 (2018)
5. Wu, Y.L., Zhao, S.L., Li, C.J.: Text classification method based on TF-IDF and cosine similarity. J. Chin. Inform. Process. **31**(5), 138–145 (2017)
6. Yao, Q.Z., Song, Z.L., Peng, C.: Research on text categorization based on LDA. Comput. Eng. Appl. **47**(13), 150–153 (2011)
7. Xia, C.L., Qian, T., Ji, D.H.: Event convolutional feature based news documents classification. Appl. Res. Comput. **4**, 991–994 (2017)
8. Zhang, T.: A kind of effective data aggregating method based on compressive sensing for wireless sensor network. EURASIP J. Wirel. Commun. Netw. **2018**(159), 1–15 (2018). https://doi.org/10.1186/s13638-018-1176-4
9. Zhang, D.G., Zhang, T.: Novel optimized link state routing protocol based on quantum genetic strategy for mobile learning. J. Netw. Comput. Appl. **122**, 37–49 (2018). https://doi.org/10.1016/j.jnca.2018.07.018
10. Zhou, F.Y., Jin, L.P., Dong, J.: Review of convolutional neural network. Chin. J. Comput. **1**, 35–38 (2017)
11. Li, Y., Dong, H.B.: Text emotion analysis based on CNN and BiLSTM network feature fusion. Comput. Appl. **38**(11), 29–34 (2018)
12. Kalchbrenner, N., Blunsom, P.: Recurrent convolutional neural networks for discourse compositionality. Comput. Sci. **10**, 1–2 (2013)
13. Kim, K., Chung, B.S., Choi, Y.R.: Language independent semantic kernels for short-text classification. Expert Syst. Appl. Int. J. **41**(2), 735–743 (2014)
14. Liu, S.: Novel unequal clustering routing protocol considering based on network partition & distance for mobile education. J. Netw. Comput. Appl. **88**(15), 1–9 (2017). https://doi.org/10.1016/j.jnca.2017.03.025
15. Zhou, S.: A low duty cycle efficient MAC protocol based on self-adaption and predictive strategy. Mob. Netw. Appl. **23**(4), 828–839 (2018)
16. Jin, C., Li, W., et al.: Chinese word segmentation based on bidirectional LSTM neural network model. Chin. J. Inform. **32**(2), 29–37 (2018)
17. Chen, J., Li, H.F., Ma, L., et al.: Dimensional speech emotion recognition method based on multi-granularity feature fusion. Signal Process. **33**(3), 374–382 (2017)
18. Zhang, D.G., Niu, H.L., Liu, S.: Novel PEECR-based clustering routing approach. Soft. Comput. **21**(24), 7313–7323 (2017)
19. Tang, Y.M.: Novel reliable routing method for engineering of internet of vehicles based on graph theory. Eng. Comput. **36**(1), 226–247 (2019)
20. Fan, Y.X., Guo, J.F., Lan, Y.Y., et al.: Context-based deep semantic sentence retrieval model. Chin. J. Inform. Sci. **31**(5), 161–167 (2017)
21. Hatzivassiloglou, V., Mc Keown, K.R.: Predicting the semantic orientation of adjectives In: Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 174–181 (1997)
22. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Acl-02 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 79–86 (2002)
23. Turney, P.D.: Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In: Proceedings of Annual Meeting of the Association for Computational Linguistics, pp. 417–424 (2002)
24. Lin, C., He, Y., Everson, R.: A comparative study of Bayesian models for unsupervised sentiment detection. GRB Coord. Netw. **12255**, 144–152 (2010)
25. Zhang, D.G., Chen, C., Yu, Y.C., et al.: New method of energy efficient subcarrier allocation based on evolutionary game theory. Mob. Netw. Appl. **9**, 1–15 (2018). https://doi.org/10.1007/s11036-018-1123-y
26. Liu, S.: Novel dynamic source routing protocol (DSR) based on genetic algorithm-bacterial foraging optimization (GA-BFO). Int. J. Commun Syst. **31**(18), 1–20 (2018). https://doi.org/10.1002/dac.3824
27. Simonyan, K., Zisserman, A.: Two-Stream Convolutional Networks for Action Recognition in Videos. University of Oxford, Oxford (2014)
28. Zhang, T.: Novel self-adaptive routing service algorithm for application of VANET. Appl. Intell. **49**(5), 1866–1879 (2019). https://doi.org/10.1007/s10489-018-1368-y
29. Zhang, D.G., Hui, G.: New multi-hop clustering algorithm for vehicular ad hoc networks. IEEE Trans. Intell. Transp. Syst. **20**(4), 1517–1530 (2019)
30. Liu, S.: Dynamic analysis for the average shortest path length of mobile ad hoc networks under random failure scenarios. IEEE Access. **7**, 21343–21358 (2019). https://doi.org/10.1109/ACCESS.2019.2896699
31. Gao, J.X.: Novel approach of distributed & adaptive trust metrics for MANET. Wirel. Netw. **25**(6), 3587–3603 (2019)
32. Zhang, T.: A kind of novel method of power allocation with limited cross-tier interference for CRN. IEEE Access. **7**(1),

82571–82583 (2019). https://doi.org/10.1109/ACCESS.2019.2921310
33. Hermans, M., Burm, M., Dambre, J.: Trainable and dynamic computing: error backpropagation through physical media. Arxiv **1**, 34–39 (2014)
34. Otte, S., Krechel, D., Liwicki, M. et al.: Local Feature Based Online Mode Detection with Recurrent Neural Networks In: International Conference on Frontiers in Handwriting Recognition. IEEE Computer Society, pp. 55–60 (2012)
35. Pérez, Z., Cardona-Escobar, A.F.: Deep Convolutional Neural Networks and Power Spectral Density Features for Motor Imagery Classification of EEG Signals In: International Conference on Augmented Cognition. Springer, Cham, pp. 99–106 (2018)
36. Zeng, Y., Ferdous, Z.I., Zhang, W., et al.: Inference with hybrid bio-hardware neural networks. **2019**(5), pp. 57–61
37. Wang, Y., Zhang, B., Xue, B.: Research on Chinese text classification method based on FOA-SVM. J. Sichuan Univ. (Nat. Sci. Ed.) **53**(4), 101–104 (2016)
38. Gao, J.P., Zhang, H., Zhao, X.J., et al.: Research on WEB domain knowledge classification based on feature words. Softw. Guide **15**(2), 9–11 (2016)

Zi-xian Liu Born in 1998. Researcher, Computer Science and Technology, Tianjin University of Technology, Tianjin, 300384, China. His research interest includes User Portrait, Natural Language Processing, etc.

De-gan Zhang (M'01) Born in 1970, Ph.D. Graduated from Northeastern University, China. Now he is a professor of Tianjin Key Lab of Intelligent Computing and Novel software Technology, Key Lab of Computer Vision and System, Ministry of Education, Tianjin University of Technology, Tianjin, 300384, China. His research interest includes service computing, etc.

Gu-zhao Luo Born in 1999. Researcher, Computer Science and Technology, Tianjin University of Technology, Tianjin, 300384, China. His research interest includes User Portrait, Natural Language Processing, etc.

Ming Lian Born in 1998. Researcher. He is certificated as an Engineer of Information Security, and now he is a senior in Tianjin University of Technology, Tianjin, 300384, China. His research interest includes Information Security, Machine Learning, etc.

Bing Liu (M'01) Born in 1966, MA. Eng Graduated from Tianjin University of Technology, China. Now he is a professor of Department of Computer Technology, School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, 300384, China. And he is also the teaching supervisor of the school. At present, his research fields mainly include Computer Network Technology, Multimedia Technology, Database Technology, etc.