IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 17, 2021, accepted October 28, 2021, date of publication November 1, 2021, date of current version December 20, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3124585

Individual Emotion Recognition Approach Combined Gated Recurrent Unit With Emoticon Distribution Model

CHANG LIU[1], TAIAO LIU[2], SHUOJUE YANG[3], AND YAJUN DU[2]
[1]Chengdu Ruibei Yingte Information Technology Company Ltd., Chengdu 610094, China
[2]School of Computer and Software Engineering, Xihua University, Chengdu 610039, China
[3]Department of Mechanical Engineering, School of Engineering, Johns Hopkins University, Baltimore, MD 21218, USA

Corresponding author: Yajun Du (duyajun@mail.xhu.edu.cm)

This work was supported in part by the National Natural Science Foundation under Grant 61872298 and Grant 61802316, and in part by the Science and Technology Department of Sichuan Province under Grant 2021YFQ0008.

ABSTRACT Mining individual emotions from the data is extremely challenging work in many fields, such as opinion monitoring, business decisions, and information prediction. In previous studies, many studies recognized texts containing positive, negative and neutral sentiments, but few studies focus on emotions according to multiple categories, such as happiness, sadness, anger, fear, disgust and surprise. In this paper, we propose an individual emotion-identifying model called semantic emoticon emotion recognition (SEER). First, we divide the input short text into four categories by using the emotion dictionary and emoticons. Second, we combine a bidirectional gated recurrent unit (Bi-GRU) network with an attention mechanism to capture the emotion vectors of input text from the word aspect. Third, we construct an emoticon distribution model to obtain the emotion vectors in a large quantity of social network data. Fourth, according to different types of input short texts, we select different fusion weights to fuse the emoticon emotion features in text and the semantic emotion features of the texts. Finally, we classify the short text emotions into six categories depending on the final emotion vector. The experimental results show that the SEER is an effective method for improving the accuracy of emotion recognition. In addition, our proposed approach achieves the highest accuracy of 86.35% in emotion recognition, which indicates that the accuracy of our proposed approach achieves improvements of 12.66%, 14.56%, 5.38%, 4.48%, 4.9%, 2.68%, and 3.17% compared to SVM, WL, LSTM, BiLSTM, GRU, Bi-GRU, and CNN+BiLSTM, respectively.

INDEX TERMS Emotion recognition, bidirectional gated recurrent unit, attention mechanism, emotion distribution model.

I. INTRODUCTION

With The rapid development of the internet, the popularity of computers, universalized smartphones, and different kinds of social media platforms, such as Weibo, QQ, WeChat, Facebook, and Instagram, can provide convenient information exchange services. An increasing number of people are willing to post text information to express their emotions anywhere and anytime on social networks. Due to the massive posted short text information, some researchers have prompted the analysis of emotion in short texts during the last decade [1]. Emotion takes on very complex multidimensional features and is also a key factor that affects people's behavior. In addition, emotional analysis is called emotional computing, which helps merchants effectively develop sales strategies by analyzing product reviews and helps politicians improve national policy. Furthermore, emotional analysis has good application prospects in public opinion monitoring, business decisions, information prediction, and even presidential elections. In recent years, emotion analysis has become a popular trend in social media, and a large number of researchers have studied and explored it.

An Oxford dictionary defines 'emotion' as "a strong feeling deriving from one's circumstances, mood, or relationships with others" [2]. For example, you may feel sad when your friends are uncomfortable or you feel happy when you achieve goals. The American Psychological Association defines 'emotion' as "a complex pattern of changes,

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang.

including physiological arousal, feelings, cognitive processes, and behavioral reactions'' [3]. In general, emotion is a response to external stimuli, which is a physical and psychological state resulting from multiple feelings, thoughts, and behaviors. In 1992, American psychologist Ekman [4] proposed a basic theory of emotions. In this theory, the basic emotions include happiness, sadness, anger, fear, disgust, and surprise, which can be distinguished by facial expressions and physiological processes. To better understand individual emotional states, we use deep learning, attention mechanisms and emoticon distribution models to divide emotions into six classes based on Ekman's emotion category theory. In the past few years, researchers have mostly relied on emotional words that are defined in an emotion knowledge base (WordNet Affect, LIWC) [5] to recognize emotion. However, these methods need a high-quality knowledge base to guarantee the accuracy of emotion recognition. Obviously, it is not easy to identify new emotional words or slang in social networks in time. Therefore, it is difficult to establish a high-quality knowledge base for emotion classification. Hence, the previous lexicon-based methods for emotion recognition did not work well. Moreover, emotion recognition from short text has many challenges to overcome; for instance, emotions contained in short texts are expressed implicitly, and the text includes one or more kinds of emoticons. For these challenges, we leverage the Bi-GRU network to capture the emotion features of the words aspect and utilize the emoticons distribution model to obtain the emoticon emotion features.

However, the traditional methods used for emotion analysis are limited because they cannot deal with large numbers of short texts simultaneously, cannot effectively capture implicit emotional features, and do not consider emoticons in emotion analysis tasks. In recent years, deep learning has been increasingly applied in neural language processing, including emotion analysis. In particular, RNN can flexibly process sentences of different lengths and reform any sentence lengths to a floating-point number vector of a specific dimension. However, it cannot capture long-term dependencies well and has the problem of gradient disappearance, which causes a negative impact on emotion recognition. To solve the problem, LSTM [6], which is a special kind of RNN that can learn long-term dependencies, has been posited. LSTM has been refined and popularized by many researchers in the following works. In 2014, Cho *et al.* [7] proposed a GRU based on the LSTM, which can also solve the long-term dependence problem in RNNs. In addition, the GRU has similar accuracy to LSTM, and the computation is less expensive than LSTM.

Based on the GRU, we propose a novel emotion recognition model called ''semantic and emoticon-based emotion recognition (SEER)'' to recognize emotions from the input sentence. In this model, we use Bi-GRU to capture semantic features in the word aspect. We use the emoticon distribution model to analyze the emoticon emotion contained in a sentence. Furthermore, we utilize a self-attention mechanism to obtain the weight of the semantic features in the word aspect after the Bi-GRU layer. Then, the emoticon aspect is used to adjust the emotion vectors to improve the model accuracy. We evaluate SEER on real-world textual data, and it outperforms traditional machine learning approaches and other deep learning approaches. We propose an individual emotion-identifying model called semantic emoticon emotion recognition (SEER), and our main contributions are as follows:

- We divide the input short texts into four categories, including explicit emotional word with emoticon, implicit emotional word with emoticon, only emoticon and only word, by using the emotion dictionary and emoticons. We combine a bidirectional gate recurrent unit (Bi-GRU) network with an attention mechanism to capture the emotion vectors of input text from the word aspect.
- We construct an emoticon distribution model to obtain the emotion vectors in a large quantity of social network data. According to different types of input short texts, we select different fusion weights to fuse the emoticon emotion features in text and the semantic emotion features of the texts. We classify the emotions of short texts into six categories depending on the final emotion vector.
- The experimental results show that the SEER is an effective method for improving emotion recognition accuracy. In addition, our proposed approach achieves the highest accuracy of 86.35% in emotion recognition, which indicates that the accuracy of our proposed approach achieves improvements of 12.66%, 14.56%, 5.38%, 4.48%, 4.9%, 2.68%, and 3.17% compared to SVM, WL, LSTM, BiLSTM, GRU, Bi-GRU, and CNN+BiLSTM, respectively.

The structure of this paper is as follows. Section 2 reviews the related work on emotion recognition; Section 3 describes in detail our model for emotion analysis from short texts. Section 4 describes the datasets used in our evaluation and the baseline methods for the comparative analysis. Section 5 illustrates the experimental results and the parameter setting. Finally, Section 6 presents conclusions and future work.

II. RELATED WORK

As emotion is a natural human attribute that constantly affects people's behavior. If they perform some bad behaviors due to emotions, it is necessary to find a method to recognize human emotions. Emotion analysis from text mainly includes two aspects of research content: text emotion recognition and text emotion classification. Text emotion classification is also based on text emotion recognition research. The task of text emotion classification is to divide the text into two or more predefined emotion categories by capturing the emotional elements in the text content. To date, the emotion classification research methods include supervised learning methods [8]–[10], dictionary-based methods, machine learning methods, unsupervised learning methods [11], deep learning methods, and hybrid approaches.

A. DICTIONARY-BASED METHODS

In previous research, Desmet and Hoste [12] hypothesized that lexical and semantic features can represent the data information and use binary support vector machine classifiers to detect emotion for suicide prevention. Park *et al.* [13] developed a method to capture the depressive emotion of Twitter users. To determine how users' depressive emotions affect their behaviors, they analyzed 69 individual texts. In general lexicon-based strategies, Joshi *et al.* [14] used an associated emotion lexicon to find the right emotion from some texts. Wu *et al.* [15] considered that each emotional word can express different emotions in different sentences because the emotion conveyed by the same word sometimes changes with the context. They constructed word lexicons by the co-occurrence of word and sentence labels and used the average emotion intensity to represent the emotion category. Therefore, they focused on emotion recognition from sentences by using constructed word lexicons. Although the dictionary-based method is easy to understand, it also requires much manual work to build the emotional dictionary, and the effect completely depends on the quality of the emotional dictionary.

B. MACHINE LEARNING METHODS

To overcome the shortcomings of dictionary-based methods, researchers began to utilize machine learning to recognize emotion after extracting features from the text. Machine learning strategies, which apply numerous existing classifications and clusters, adopt supervised and unsupervised learning to emotion detection. Mohammad and Kiritchenko [16] selected support vector machines (SVMs) with sequential minimal optimization [17] as the machine learning algorithmic rule because they are effective in various analysis issues. They proposed a novel approach that can automatically classify Twitter messages to show a user's emotional status. To model emotional status, they used the well-established model, which contains two dimensions: valence and arousal. Additionally, the importance of unsupervised learning is particularly increasing in the context of the fact that massive data can be easily obtained but is difficult to label, and the proportion of unstructured data continues to increase. Agrawal and An [18] proposed a completely unique unsupervised context-based approach to recognize emotion from sentences. The method does not rely on any existing lexicons (WordNet Affect). Furthermore, the methods based on the linguistic connection between words and the varied emotions build emotion vectors for every word that indicate human emotions. Based on this method, Hajar *et al.* [11] reported another unsupervised machine learning algorithm to classify emotions in YouTube reviews. They employed the pointwise mutual information (PMI) measure to calculate its similarity with every target emotion for classifying texts into a defined emotion classification. Normal machine learning methods, such as naive Bayes [19], maximum entropy, and support vector machine [20], are usually used in emotion classification tasks. Machine learning methods have strong generalization capabilities, but training is time-consuming and sensitive to the choice of parameters and kernel functions.

C. DEEP LEARNING METHODS

Recently, deep learning and attention mechanisms have also been employed by researchers in several studies. Johnson and Zhang [21] directly used the convolutional neural network (CNN) and learned embedding on high-dimensional text data for text categorization. Su *et al.* [22] introduced LSTM to analyze emotion from the NLPCC-MHMC-TE database. In this study, they used word2vec to obtain semantic word vectors and applied an affective lexicon to obtain the emotional word vector. Then, they used two kinds of vectors as LSTM model inputs to classify the emotion into seven categories. Tai *et al.* [23] proposed a tree-LSTM model that is a standardization of LSTM (long short-term memory). In this study, every tree-LSTM unit includes an input gate, output gate, and hidden state. Moreover, updating cells depended on other states of the child unit. To mix information from each child effectively, the tree-LSTM has one forget gate for every child. After Bahdanau *et al.* [24] introduced attention mechanisms to the field of natural language processing, it elicited wide popularity in utilizing attention mechanisms in many NLP tasks, such as RNNs and CNN models. Li *et al.* [25] advanced a multi attention-based neural network to analyze the emotional cause. Their main attention mechanism work was to capture the interaction between the emotion clause and every candidate clause. It helps the convolutional neural network build the emotion cause clause. Although RNN can handle the problem of text timing well, it cannot solve the problem of long-term dependence.

D. HYBRID APPROACH

Machine learning has received attention in short text emotion analysis, and the hybrid approach has also caught researchers' attention. The best advantage of a hybrid approach is to combine different methods to use each method's merits and minimize each method's drawbacks to analyzing emotion contained in short texts. Wu *et al.* [26] mixed machine learning and knowledge-based methods to recognize emotions at the sentence level. In this study, depending on emotion generation rules (EGRs), the authors use semantic labels (SLs) and attributes (ATTs) to represent the emotion of each sentence. Then, SLs and ATTs consist of emotion association rules (EARs). Finally, to calculate the similarity between the input sentence and emotion association rules of every emotion, the authors adopt a separable mixture model (SMM). Recently, Perikos and Hatzilygeroudis [27] used the naive Bayes (NB) classifier and maximum entropy learner to identify emotion in short text and adopted knowledge-based methods to deeply analyze language structure. After each classifier had an output, they used a majority voting approach [28] to create an emotion classification decision. Rajabi *et al.* [29] proposed a hybrid approach that used the LSTM to learn the sequential aspect of the data, and the CNN was used to extract the fine-grained features. They then combined the CNN with

TABLE 1. The advantages and disadvantages of different methods.

Method
Word Lexicons(WL) Advantage: understandable and simplified models. Disadvantage: heavy manual workload
Support Vector Machine(SVM) Advantage: strong generalization ability. Disadvantage: training is time-consuming, sensitive to parameter and kernel function selection
Recurrent Neural Network(RNN) Advantage: solve timing issues. Disadvantage: long-term dependence and vanishing gradient
Long Short-Term Memory/Gated Recurrent Unit(LSTM/GRU) Advantage: long-term memory. Disadvantage: cannot parallel training and not friendly to transfer learning

the BiLSTM to classify emotion based on these short texts. To obtain better recognition performance, Sun *et al.* [30] proposed a hybrid users' emotion detecting model combining CNN and LSTM (CNN-LSTM) with a Markov chain Monte Carlo (MCMC).

In addition, some approaches [31] can recognize ternary emotion (positive, neutral, negative) by using transfer learning. However, They did not focus on six element emotion (happiness, sadness, anger, fear, disgust, and surprise) recognition.

In our daily life, communication on social networks can be expressed not only in words, phrases, sentences but also in pictures and emoticons [32]. It is essential to consider emoticons in emotion recognition. Jiang *et al.* [33] proposed an ESM model that includes projection and classification phases. This model builds the emoticon space with many emoticons. Then, it uses the feature vectors to perform the supervised sentiment classification tasks. After considering a large amount of previous related work, we propose the SEER method. We first analyze the text and emoticons separately and then merge the emotional features of sentences and emoticons for emotion recognition. Compared with other methods, our method can learn automatically without much manual work, deal with text timing problems, solve long-term dependence problems, and handle emoticons better.

In summary, the advantages and disadvantages of different methods can be found in Table 1.

III. EMOTION RECOGNITION

A. OVERVIEW

In this section, we briefly describe the RNN and LSTM, and then we introduce the Bi-GRU network. Recurrent neural networks are based on feedforward neural networks that have internal memory, but unlike feedforward neural networks, RNNs have the capacity to utilize their internal memory to process sequences of inputs. This makes RNN applicable to some NLP tasks, such as machine translation [34], sentiment classification [35], or named entity recognition [36]. Similar to feedforward neural networks, RNNs also consist of an input layer, hidden layer, and output layer. However, RNNs have the same weights and biases in the hidden layer. Therefore, we can provide input to the hidden layer at each step. In addition, a recurrent neuron will keep the input information of all the previous steps with the input of the current step. Thus, it also captures some information regarding the correlation between the current and previous steps, and the decision at the $t-1$ step affects the decision at step t. This is extremely analogous to how humans make decisions. However, there are two obviously fatal flaws in RNNs: vanishing and exploding gradients.

In 1997, Hochreiter and Schmidhuber [6] developed another deep learning language model (LSTM) that can effectively overcome the RNN shortcomings. The memory block with a single LSTM cell in [37] can illustrate how the LSTM works. The LSTM neural network consists of three gates: an input gate, a forget gate, and an output gate. These gates protect and control the cell states. That is why the LSTM can deal with long-distance history information. In the LSTM model, the forget gate, which is located at first a sigmoid layer, decides what information should be "forgotten". Then, the LSTM model uses the input gate located in another sigmoid layer to decide what information should be stored in the cell states. Finally, based on these cell states, the output gate located in a sigmoid layer decides what will be output. The output numbers of all sigmoid layers lie in [0, 1]. It describes how much of each piece of information should be forgotten, input, and output. When the output number is closer to one, it indicates that more information can pass through the gates.

B. OUR PROPOSED MODEL

In this section, we define our problem as a multiclass recognition task that classifies the emotions contained in short text into happiness, sadness, anger, fear, disgust, and surprise. To precisely identify emotions from short text, we propose a novel model called SEER (semantic emotion recognition), which includes a word embedding layer, a Bi-GRU neural network layer, an attention mechanism layer, emoticon embedding, a connection layer, a softmax layer, and a sentence classification layer. The architecture of our model, which is based on our initial contribution in [38], is illustrated in Fig. 1. In this novel model, it consists of seven parts:

- **Sentence Classification**: The sentence classification layer is used to divide the sentence into four categories: sentences that contain explicit emotional words, sentences that contain explicit emotional words and emoticons, sentences that have implicit emotional words, and sentences that have implicit emotional words and emoticons.
- **Word Embedding Layer** : The word embedding layer transforms the input sentence into a word vector that can be used as the input for the next layer.
- **BI-GRULayer**: The Bi-GRU network extracts semantic features from the word aspect.
- **Attention Layer**: This layer utilizes a self-attention layer to give the different weights for each word.
- **Emoticon Embedding**: It uses the emoticon embedding model to recover the emotion contained in the emoticons aspect.

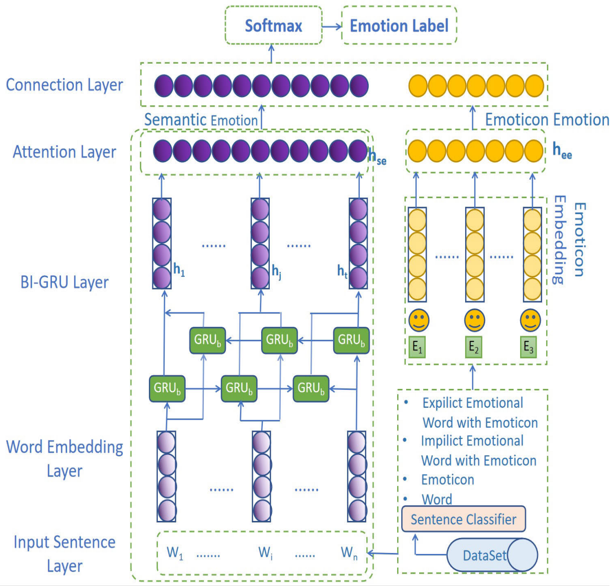


FIGURE 1. SEER model architecture.

- **Connection Layer**: This layer utilizes a connection layer to connect the semantic and emoticon features.
- **Emotion Label**: It uses the outputs of the connection layer used for recognizing emotions via the softmax function.

C. SENTENCE CLASSIFICATION LAYER

After analyzing the sentences in the datasets, we find that some sentences have explicit emotional words and emoticons, some have only explicit emotional words or only emoticons, and some contain implicit emotional words and emoticons. Therefore, we classify the sentences based on the sentiment dictionary and emoticons set in the sentence classification layer to train our proposed model better. We classify sentences into four categories: sentences that only have explicit emotional words without emoticons, sentences that not only have explicit emotional words but also emoticons, sentences that only have emoticons, and sentences that have emoticon and implicit emotion words. For example, the example sentences in Table 2 are composed of implicit emotion words and emoticons, and the example sentence in Table 3 consists of explicit emotional words and emoticons. The purpose of this layer is to train our model on different sentence categories to obtain the most suitable p in the fusion layer. p can be considered the dynamic weight in the feature fusion layer, and the value of p is different under each sentence category. In the experimental part, the influence of the effect of our model under different values of p will be discussed.

D. WORD EMBEDDING LAYER

Word embedding is the first step and essential part of natural language processing. Before using Bi-GRU to extract semantic emotion from the text, we need to obtain the vector representation of the input words. The input sentence is transformed into a word vector belonging to a d-dimensional vector by word2vec, which is well trained on Wikipedia data. We denote S_{sen} (Eq. (1)) to the input sentence consisting of T words, and W_i (Eq. (2)) denotes the word vector of the $i-th$ word. The matrix S_{sen} represents the output of the word embedding layer, and each row in the matrix represents the semantic vectors of each word, which is also utilized as the input of the Bi-GRU layer.

$$S_{sen} = [W_1, \ldots, W_i, \ldots, W_T] \in R^{T\times d} \tag{1}$$

$$W_i = [m_{i1}, \ldots, m_{ik}, \ldots, m_{id}] \tag{2}$$

E. BI-GRU LAYER

The gated recurrent unit GRU based on LSTM was proposed by Cho in 2014 [7]. The memory cell in the GRU is removed, and the gate is changed into a reset gate and update gate. Compared with LSTM, GRU performs slightly better than LSTM in the sentiment classification task, and GRU is computationally less expensive [39]. To better understand the semantic information of the text, we employ bidirectional GRU to extract emotion features. After we obtain the input of the Bi-GRU layer from the word embedding layer, we can initially extract the emotional features of the words. The process of emotional feature extraction and the single cell of Bi-GRU are shown in Fig. 2.

The text emotion feature extraction of the GRU neural network can be summarized as follows:

- Use the useful information of the previous state h_{t-1} and the input of the current state x_t to calculate the update gate (U_t) and reset gate (R_t). The calculation methods of R_t and U_t are shown in Eq. (3) and Eq. (4), respectively.
- Then, obtain the new hidden state of the current cell h_t according to the update gate and reset gate, and the calculation process of h_t is from Eq. (5), Eq. (6) to Eq.(7), where W_R, W_U, and W_h, in these formulas from Eq. (3) to Eq. (7) are parameters that need to be learned.

First, we extract the emotion features of the current word based on the inputs x_t and h_{t-1}, where x_t represents the embedding vector of the current word, and h_{t-1} denotes the hidden state passed from the previous cell. It contains some important information about the previous word.

Second, the *sigmoid* function is used as the reset gate (R_t), which is calculated by Eq. (3) and update gate (U_t) is calculated by Eq. (4) to reset and update information from x_t and h_{t-1}. The reason why we use the *sigmoid* function is that the output value of the *sigmoid* function is from 0 to 1. The closer the value is to 1, the more the information from x_t and h_{t-1} is reset and updated.

Specifically, when R_t is used to reset h_{t-1}, we use h'_{t-1} to represent the reset information, which is calculated by Eq.(5). U_t is used to update the information of x_t and h_{t-1}, but before utilizing U_t, the *tanh* function was previously used to address x_t. The output of *tanh* function shown in Eq. (6) denotes h'_t, which denotes the candidate hidden state, and the main information contained in h'_t is the current word.

Ultimately, the information is updated by Eq. (7), and h_t, the hidden state of the current cell, is obtained so that the

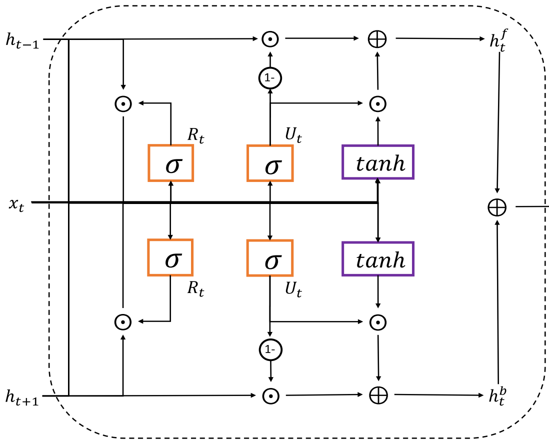


FIGURE 2. The structural diagram of the Bi-GRU cell.

output of the current cell is also passed to the next cell, where W_R, W_U, and W_h in these formulas from Eq. (3) to Eq. (7) are parameters that need to be learned.

$$R_t = Sigmoid(W_R[h_{t-1}, x_t]) \tag{3}$$
$$U_t = Sigmoid(W_U[h_{t-1}, x_t]) \tag{4}$$
$$h'_{t-1} = h_{t-1} \odot R_t \tag{5}$$
$$h'_t = tanh(W_h[h'_{t-1}, x_t]) \tag{6}$$
$$h_t = U_t \odot h_{t-1} + (1 - U_t) \odot h'_t \tag{7}$$

For the purpose of using the left and right context of the current word more effectively, the Bi-GRU network extracts sentence emotion features not only from the forward to backward direction but also from the reverse direction. The symbols GRU_f and GRU_b express the forward GRU and backward GRU, respectively, and h_t^f and h_t^b denote the hidden states in the forward GRU and backward GRU, respectively, which are calculated via Eq. (8) and Eq.(9). After we obtain h_t^f and h_t^b from each input word through the Bi-GRU network, we can further obtain the final output of the Bi-GRU network to connect the two hidden states by using Eq. (10).

$$h_t^f = GRU_f(w_i, h_{t-1}^f) \tag{8}$$
$$h_t^b = GRU_b(w_i, h_{t+1}^b) \tag{9}$$
$$h_t = [h_t^f, h_t^b] \tag{10}$$

F. SELF-ATTENTION LAYER

In the input sentence, different words play different roles in emotion classification. Place and time adverbial clauses have little importance for emotion classification. Adjective, verb, and noun words are very important. To highlight the importance of different words in the emotion identification of sentences, an attention layer after the Bi-GRU layer is utilized to further extract precise emotion features and to give the different weights for each word. More important words can obtain higher weights, and less important words obtain lower weights. For example, in the sentence "It is a sunshine day but I have to stay at home, so sad", the positive word "sunshine" is not useful information for emotion recognition but the words "stay" and "sad" are the important words on which we should place higher attention. From previous research [40], we know that there are many kinds of attention mechanisms applied in the field of natural language processing. We select the most suitable self-attention mechanism in the attention layer.

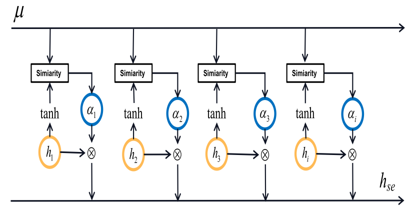


FIGURE 3. The attention mechanism process.

The detailed process of the self-attention mechanism is described in Fig. 3. The input of the attention layer is h_t, which stems from the output of the Bi-GRU layer. First, h_t uses the *tanh* function to obtain the hidden representation θ_i in the attention layer. The *tanh* function expresses by Eq.(11). Second, we initialize a query vector μ randomly and then utilize the *softmax* function to calculate the similarity between μ and θ. The *softmax* function is shown in Eq.(12). α_i represents the attention distribution, which represents the importance of word i. After obtaining the attention distribution for each word, Eq. (13), which averages all the input word vectors with the attention distribution, can be used to calculate the output of the attention layer, where we denote h_{se} as the output, and h_{se} also represents the semantic emotion vector in the word aspect of the input sentence.

$$\theta_i = tanh(W_m h_t + b_m) \tag{11}$$
$$\alpha_i = \frac{exp(\theta_i^T \mu)}{\sum_{i=1}^{T} exp(\theta_i^T \mu)} \tag{12}$$
$$h_{se} = \sum_{i=1}^{T} \alpha_i h_t \tag{13}$$

G. EMOTICON EMBEDDING LAYER

The emoticon plays an essential role in emotion recognition because emoticons include the real emotions embedded in the text. Moreover, emoticons can express emotions more directly. From some datasets, we can observe the phenomenon that people often express emotion implicitly without any emotional words but with emoticons. For example, in Table 2, if we read the sentence "Rainy days are better for chocolate" from the word aspect, we cannot identify what emotion is desirable to express in this sentence. However, depending on the emoticon "😀", we can easily recognize that the emotion of this sentence is "happy". Psychologically speaking, emoticons express users' emotions more visually and intuitively. Therefore, emoticons in tweets are becoming increasingly frequent, and emoticons have become

TABLE 2. The example of some emoticons in text.

Example tweets	Emotion	Emoticon	Tweets
雨天和巧克力更配哦。😀 Rainy day is better for chocolate.	Neutral	Happy	Happy
又遇到这个问题。😡 Meet this problem again.	Neutral	Sad	Sad

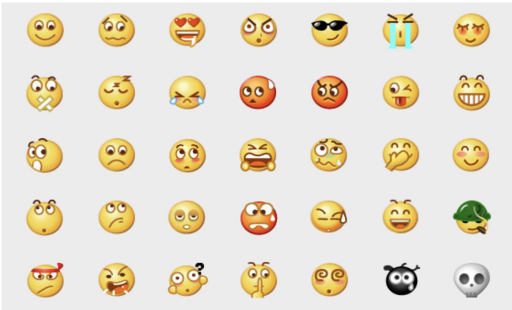


FIGURE 4. Example of emoticons in Weibo.

increasingly important in emotion recognition. Hence, we utilize emoticon distribution to extract emoticon emotion to enhance the emotion vectors and further complete the emotional classification work.

For the purpose of helping users express their emotions better, various social platforms provide a large number of emoticons for users [41]. Users can express any kind of emotion by emoticons, such as "happy" "😀", "sad" "😭", and "angry" "😡". There are some examples of emoticons in Fig. 4. However, many previous types of studies filter emoticons as noisy components or treat emoticons as the same as sentences during the emotion recognition task. In [42], Balomenos *et al.* used EmojiNet to translate an emoji into a sentence that can describe the emoji, extract emoji features through the bag-of-words model, and then combine the aspect of the words to perform the classification task. However, in our model, to avoid missing important emotional features and improve the accuracy from emoticons, we deal with emoticons and words. We use the Bi-GRU neural network to extract semantic features from the word aspect and obtain the emotional features of emoticons through emoticon distribution.

Therefore, in the emoticons distribution, we perform a statistical analysis on the NLPCC2013 and NLPCC2014 datasets. In the two datasets, we find 5,400 sentences in total, which include emoticon expressions. Some sentence samples from the two datasets are shown in Table 2. After analyzing these 5,400 sentences, we can learn about the distribution of emoticons in these sentences and obtain the emotion features of the emoticons that are used in these two datasets. For example, emoticon "😀" appears 1,125 times over all sentences, and the word "happy" appears 875 times, "surprise" appears 210 times, "sad" 14 times and "anger" 26 times. After we obtain these numbers, we use the statistical function shown in Eq. (14) to calculate the emotion features of the emoticon "😀". We use e_i to express one emoticon from the emoticon set, and J represents the emotion set in our research. We apply six different emotions and denote E_{di} in Eq. (15) as the emotion distribution of emoticon e_i. Therefore, we can obtain the emotion feature of every emoticon and express the emotion feature of each emoticon as a six-dimensional vector. For instance, the emotion feature of the emoticon "😀" can be expressed as "😀=[0.78 0.01 0.02 0 0 0.19]". These numbers successively indicate the possibility that this emoticon belongs to "happy", "sad", "angry", "afraid", "disgusted", and "surprised". Other emotion features of the emoticon sample are shown in Fig. 4.

$$E_{e_{ij}} = \frac{N_{e_{ij}}}{\sum_{j=1}^{J} N_{e_{ij}}} \tag{14}$$

$$E_{di} = [E_{e_{i1}}, E_{e_{i2}}, E_{e_{i3}}, E_{e_{i4}}, E_{e_{i5}}, E_{e_{i6}}] \tag{15}$$

where e_{ij} denotes that emoticon e_i belongs to emotion j, $N_{e_{ij}}$ denotes that emoticon e_i appears in emotion j, and obviously, $E_{e_{ij}}$ indicates how much emoticon e_i corresponds to emotion j.

We also find another important phenomenon in the two datasets. The phenomena are that the emoticon in the sentence often appears not only once but many times. Moreover, a different emoticon can appear in the same sentence. For instance, in the last example in Table 3, the emoticons "😣", "😲" and "😭" are used in the same sentence. In this case, we need to perform an extra step to determine the final emotion vector from all the emoticons in one sentence. Therefore, we sum and average the emotion vectors of all emoticons in one sentence, denote h_{ee} as the final emotion vector of the emoticon, and calculate it by Eq.(16). The m in Eq. (16) denotes the number of emoticons in one sentence.

$$h_{ee} = \frac{\sum_{n=1}^{m} E_{di}}{m} \tag{16}$$

H. VECTOR FUSION LAYER

In the vector fusion layer, after proceeding with the semantic emotion h_{se} based on the words and emoticon emotion h_{ee} from the emoticons in one sentence, we incorporate these two vectors as M in the connection layer and treat it as the input of the classification layer. The incorporation process is shown in Eq. (17), where p in this formula denotes the weight of emoticon emotion in emotion recognition. Since we divide the sentence into four classes, the value of p is different for each kind of input sentence. It is easy to understand that when there are no emotional words in the sentence, the role of the emoticon will be more important in the emotional analysis process. In contrast, the importance of emoticons will decrease when the input sentence includes explicit emotional words.

$$M = (1-p)h_{se} \oplus p h_{ee} \tag{17}$$

TABLE 3. Example sentence with emoticon in the datasets.

Example tweets in the datasets	Emotion
这声音好好听！我禁不住的鼓鼓掌。 What a lovely voice! I couldn't help clapping.	Happy
其实沉默的环境并不可怕,可怕的是沉默的心。 In fact, the silent environment is not terrible, the terrible thing is the silent heart.	Sad
好开心的一天奥特曼.... What a happy day Ultraman....	Happy
人贩子必须受到惩罚。 Human traffickers must be punished.	Anger
还听错了问题。。。杯具。。。 I heard the question wrong... oh damn!	Sad

Algorithm 1 Process of Emoticon Emotion Distribution

Input

$T = [s_1, s_2, \ldots, s_p]$;//Emotional annotation text with emoticon

$E = [e_1, e_2, \ldots, e_n]$;//Set of emoticon

Output

h_{ee};//Emotion distribution of all emoticons in one sentence

E_{di};//Emotion distribution of emoticons

Begin

$E_{di} = [E_{e_{i1}}, E_{e_{i2}}, E_{e_{i3}}, E_{e_{i4}}, E_{e_{i5}}, E_{e_{i6}}]$ and h_{ee};

For k=1,k<=n,k++

  For g=1,g<=p,g++

    Traverse sentence s_g and record m as the number of emoticons in s_g;

    If emoticon e_n in sentence s_g

      Record the emotion label of sentence s_g;

      $N_{e_{ij}}++$;//The number of emoticons e_i in emotion j;

      $N_{e_i}++$;//The number of emoticons e_i;

    EndIf

  EndFor

EndFor

For j=1,j<=6,j++

  $E_{e_{ij}} = \frac{N_{e_{ij}}}{N_{e_i}}$;

EndFor

$E_{di} = [E_{e_{i1}}, E_{e_{i2}}, E_{e_{i3}}, E_{e_{i4}}, E_{e_{i5}}, E_{e_{i6}}]$;

For i=1,i<=m,i++

  $h'_{ee} = h'_{ee} + E_{di}$;

EndFor

$h_{ee} = \frac{h'_{ee}}{m}$;

Return h_{ee};

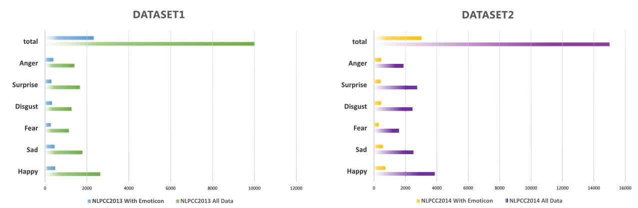
End.

I. EMOTION RECOGNITION LAYER

In this step, we use a softmax classifier to predict the emotion based on each input sentence. We define the sentence emotion as $E = [J_1, J_2, \ldots, J_j]$ and utilize Eq. (18) to calculate the probability that the input sentence belongs to the emotion classification j. Eq. (19) finds the maximum value in E as the final emotion label of the input sentence.

$$p(J_j|S_{sen}) = softmax(W_x M + b_x) \quad (18)$$

$$Y = max(E) \quad (19)$$



(a) NLPCCData2013 (b) NLPCCData2014

FIGURE 5. Datasets statistics.

IV. EXPERIMENTAL SETUP

A. DATA COLLECTION

In the experiment, we annotate two emotion datasets (NLPCCData2013 and NLPCCData2014) from the Conference on Natural Language Processing and Chinese Computing. In these two datasets, we use 80% of the data to train our proposed model and use the remaining data to test our proposed model. We also choose a part of the datasets that contains emoticons as our second dataset to confirm that the emoticon has a great impact on emotion recognition. The NLPCCData2013 dataset contains 10,000 Weibo tweets, and 2,340 of them contain emoticons. NLPCCData2014 contains 15,000 Weibo tweets, and 3,060 tweets include emoticons. The statistics of the datasets are shown in Fig. 5 (a) and (b).

B. DATA PREPROCESSING

Since all the data are from Weibo, which is generated by public users, it reasonably contains a large number of casual words, short forms, and writing mistakes. Moreover, these factors increase the noise of the input sentence and affect the effectiveness of our proposed method. For these reasons, all the input sentences from these two datasets need to be preprocessed before performing the recognition task. In this data preprocessing, we perform the following simple steps to decrease the noise in the input sentence for better performance of emotion recognition.

- The Weibo user IDs (starting with) are removed.
- The English alphabet and numbers are removed.
- The punctuation and special characters (such as "$" or "&") are also removed.

C. EVALUATION MEASURES

To evaluate our proposed method effectively and consider the uneven distribution of different emotion samples in the datasets, we use the most popular measure macroaveraging to evaluate the effectiveness of our proposed method.

Macro Precision:

$$MacroP = \sum_e \frac{TP_e}{TP_e + FP_e} \quad (20)$$

Macro Recall:

$$MacroR = \sum_e \frac{TP_e}{TP_e + FN_e} \quad (21)$$

TABLE 4. Hyperparameter setting.

Parameter	Value
Number of layer	1
Hidden layer size	250
Learning rate	0.001
Batch size	32
Dropout	0.3

Macro F-score:

$$MacroF = \frac{2 * MacroP * MacroR}{MacroP + MacroR} \tag{22}$$

where TP_e in Eqs. (20) and (21) denote the count of true positives (predicted correctly) of emotion e, FP_e expresses the number of false-positives (predicted incorrectly) of emotion e, and FN_e denotes that the number of false-positives cannot be predicted for emotion e.

D. HYPERPARAMETER SETTING

The hyperparameters need to be set in our proposed model. They include the number of layers, hidden layer sizes, batch sizes, learning rates, and dropouts. To ensure that the hyperparameters have a positive effect on emotion classification, the hyperparameters need to be studied continuously. The sets of these hyperparameters are shown in Table 4.

E. BASELINE METHODS

Word Lexicon (WL) [15]: A model that uses the word emotion lexicon to extract the emotion features from words by using emotion lexicon. We also use the emoticon features from sentences by using Bayesian classification.

Support Vector Machine (SVM) [16]: A basic support vector machine classifier based on the word vector space that is implemented to classify the emotion into six classifications.

GRU [43]: A model based on an LSTM gated network that is much simpler in structure and faster in convergence than LSTM.

Bi-GRU [44]: A model that uses the bidirectional gated recurrent unit (Bi-GRU) network with attention to classify emotion.

LSTM [45]: A traditional and basic LSTM model that uses multiclassification for text sentiment based on comments from JD.com and ctrip.com.

BiLSTM [46]: A model that analyzes implicit sentiment from text based on BiLSTM with multipolarity orthogonal attention.

CNN + BiLSTM [47]: A model that uses the CNN component to extract fine-grained features. The BiLSTM component uses the gain sequential aspect of the text and utilizes the dense layer for emotion recognition.

V. EXPERIMENTAL RESULTS AND DISCUSSION

A. EXPERIMENTAL RESULT

In this section, we evaluate our approach and describe the experimental results. We compare our proposed SEER model

TABLE 5. Comparison between SEER and baseline models on the NLPCC2013 dataset.

Model	Precision	Recall	F1-score
Word Lexicon(WL)	71.18	50.86	59.33
Support Vector Machine(SVM)	74.29	51.47	60.81
LSTM	81.07	65.59	72.51
BiLSTM	83.48	66.33	73.92
CNN+BiLSTM	84.25	67.21	74.77
GRU	80.15	65.18	71.89
Bi-GRU	83.53	67.79	74.84
SEER(proposed model)	85.76	69.09	76.53

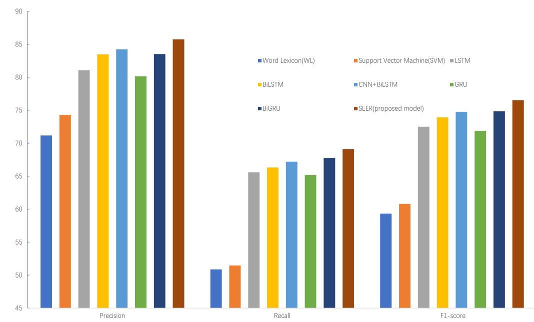


FIGURE 6. Precision, recall and F1-score comparisons between SEER and baseline models on the NLPCC2013 dataset.

TABLE 6. Comparison between SEER and baseline models on the NLPCC2014 dataset.

Model	Precision	Recall	F1-score
Word Lexicon (WL)	71.79	49.79	58.80
Support Vector Machine (SVM)	73.69	51.38	60.55
LSTM	80.97	65.34	72.32
BiLSTM	81.87	66.79	73.57
CNN+BiLSTM	83.18	67.28	74.39
GRU	81.45	64.89	72.23
Bi-GRU	83.67	67.37	74.64
SEER(proposed model)	86.35	69.83	77.22

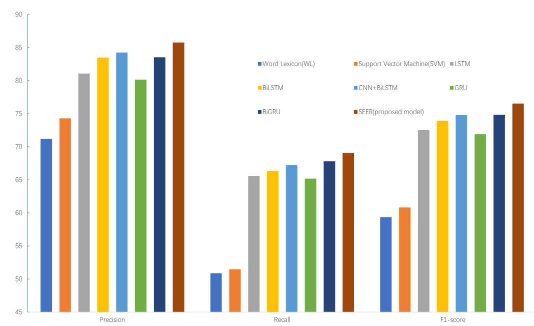


FIGURE 7. Precision, recall and F1-score comparisons between SEER and baseline models on the NLPCC2014 dataset.

with the above baselines. For all experiments, 80% of the datasets are used to train the models, and the remaining 20% of the datasets are used to test our model. Finally, we recognize the emotion by the word emotion feature and the emoticon distribution. The emotion recognition results of our proposed SEER model and the baseline model based on NLPCC2013 and NLPCC2014 datasets are shown in Table 5 and Table 6.

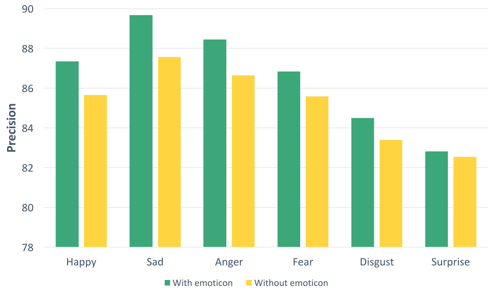


FIGURE 8. Precision comparison between emoticon and without emoticon.

As seen in Tables 5 and 6 and Figs. 6 and 7 above, we compare three different kinds of methods: lexicon-based, machine learning, and deep learning. It is proven that the machine learning method is better than the lexicon-based method, and the deep learning methods have the best performance. Compared with different deep learning methods, because the bidirectional GRU and bidirectional LSTM consider the backward information feature in recognizing emotion, the performances of Bi-GRU and BiLSTM are better than basic GRU and LSTM approaches. In addition, our proposed SEER model adds an attention mechanism and emoji features to the GRU network. Therefore, the precision, recall and F1-score are significantly better than those of the other methods. This illustrates that the attention mechanism and emoticon emotion have a positive effect on emotion recognition.

To verify the effectiveness of emoticons in emotion recognition, we train and test our proposed model on two cases (without the emoticon and with the emoticons). The effectiveness of considering emoticons in emotion classification work is shown in Fig. 8. It shows that emoticons play an important role in emotion recognition under the six different emotions. The precision of emotion recognition with emoticons is higher than that without emoticons. Moreover, the precision of emotion recognition with emoticons under happy, sad, and angry emotions is significantly higher than that without emoticons. One possible reason is that emoticons expressing happy, sad and angry emotions occur more directly and frequently than other emotions.

B. IMPACT OF PARAMETERS

In this section, we discuss the influence of the F1-score under the different values of the different parameters. In particular, we show how the six parameters impact our proposed model. The six parameters include the dimension of the hidden layer, the length of batch sizes, the value of learning rates and dropouts, the number of layers, and the value of p. First, we evaluate and compare the influence of our SEER model with other deep learning methods under different numbers of hidden layers. To guarantee the network learning ability, the numbers of hidden layers is not small. Fig. 9 (a) shows that all the methods are not sensitive to the different numbers of hidden layers except LSTM. The proposed SEER model achieves the best result when the number of hidden layers is 250. Therefore, we set 250 as the number of hidden layers in our proposed method.

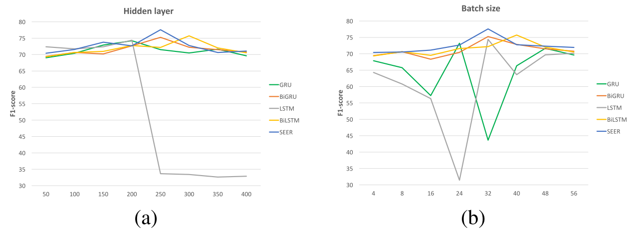


FIGURE 9. The influence of the F1-score under different hidden layer sizes and batch sizes.

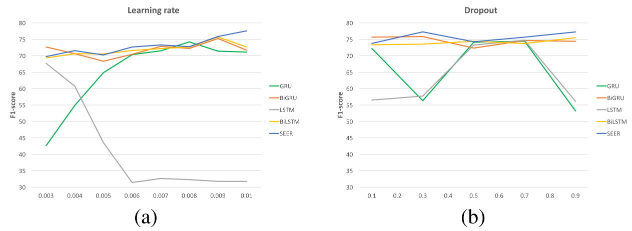


FIGURE 10. The influences of F1-scores under different learning rates and dropout.

The batch size denotes the number of selected samples for training; it is an important parameter during training models and affects the optimization and execution time of models. Then, we investigate the impact of the batch size of our SEER model and other deep learning models, and we vary the batch size from 4 to 5 or 6. Fig. 9 (b) clearly shows that the GRU and LSTM are affected easily by different batch sizes. In contrast, other deep learning methods and SEER models are not sensitive to batch sizes. In addition, the SEER model has better performance, so we set the default batch size to 32 in our proposed model.

Third, we consider the impact of the learning rate on our SEER model and other baseline models. Generally, the value of the learning rate lies between 0.0001 and 0.01. If the value of the learning rate is too small, the training time will be longer. However, if the value of the learning rate is too large, it fails to find the extreme point. Fig. 10 (a) shows that the GRU and LSTM are also sensitive to different values of learning rates. The different learning rates have almost no effect on our model. We set the default learning rate to 0.001 in our SEER model.

Dropout is another necessary parameter in the deep learning model because it can effectively prevent overfitting during the training process. We evaluate the dropout from 0.1 to 0.9 on different deep learning models. From Fig. 10 (b), we can see the effect of different dropout values on the F1-scores and find that the SEER model achieves the highest F1-score when the value of dropout is 0.3. We choose 0.3 as the default value of dropout in the SEER model.

Finally, Fig. 11 illustrates the relationship between the number of layers and the F1-score of the SEER model and other baseline deep learning methods. The GRU and LSTM are still sensitive to different numbers of layers. Thus, we set the number of layers to 1 to finish the emotion classification task.

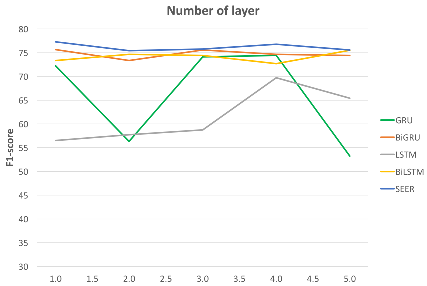


FIGURE 11. The influence of F1-scores under different layer numbers.

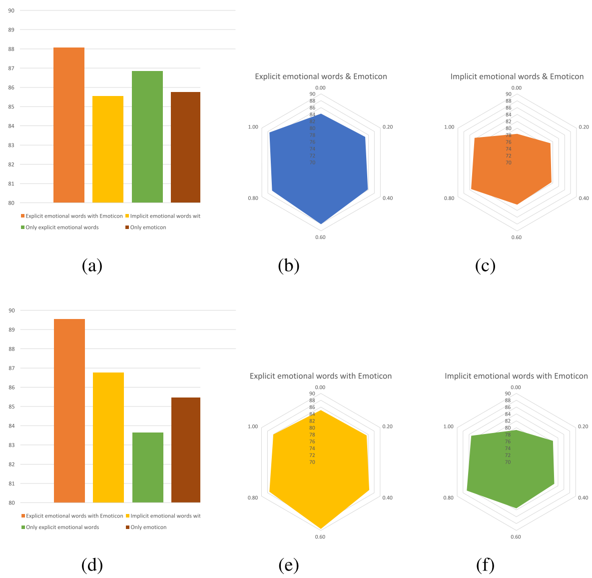


FIGURE 12. The influence of precision under different values of *p*.

To achieve a higher precision of the SEER model, we divide the sentences into four categories before word embedding and train our model in each sentence category. We use different weights p to fuse semantic emotion features and emoticon emotion features in the connection layer. Fig. 12(a), (b), and (c) are the results of our model based on NLPCC2013 data, and (d), (e), and (f) are based on NLPCC2014 data. Subgraphs (b), (c), (e), and (f) in Fig. 12 describe the change in model precision under different p values. In Fig. 12, we can see that the areas of the colored area in subgraphs (b) and (e) are larger than those of subgraphs (c) and (f). Thus, considering the sentence categories that contain explicit emotional words and emoticons, we set p to 0.6, and considering the sentence categories that contain implicit emotional words and emoticons, we set p to 0.8. Obviously, in the words-only and emoticons-only categories, the values of p are 0 and 1, respectively. The subgraphs (a), (d) in Fig. 12 illustrate the precision under different sentence categories under the two datasets. It shows that our proposed model has higher accuracy in emotion classification tasks, especially that of containing explicit emotion words in sentences.

VI. CONCLUSION AND FUTURE WORK

In our paper, it is very meaningful to develop emotion recognition of text data through artificial intelligence technology.

- We proposed an emotion recognition model called the SEER model based on text datasets.
- We used the bidirectional gated recurrent unit (Bi-GRU) network with a self-attention mechanism to capture the emotion vectors for the input data.
- We used emoticon distribution to enhance the emotion feature vectors.

In the experiment, we used comparative experiments to determine the values of the parameters in the SEER model and decide whether the attention mechanism can achieve the best performance. We compared our model with other baseline models, including word lexicon, state-of-the-art machine learning and deep learning models. From the experiment, it is shown that our model achieves better performances than other baseline models.

In future works, we will continue to construct datasets of various sizes in which each sentence contains one or more emoticons and test our SEER model. We will extend the emotional representation of emoticons, including richer emotion features, to improve our SEER model.

REFERENCES

[1] A. Giachanou and F. Crestani, "Like it or not: A survey of Twitter sentiment analysis methods," *ACM Comput. Surv.*, vol. 49, no. 2, pp. 1–41, Jun. 2016.

[2] Oxford English and Spanish Dictionary. *Meaning of Emotion in English*. Accessed: Sep. 9, 2020. [Online]. Available: https://www.lexico.com/definition/emotion

[3] American Psychological Association. *Glossary of Psychological Terms*. Accessed: Jun. 28, 2020. [Online]. Available: http://www.apa.org/research/action/glossar-y.aspx?tab=5

[4] P. Ekman, "An argument for basic emotions," *Cogn. Emotion*, vol. 6, nos. 3–4, pp. 169–200, 1992.

[5] F. Carvalho, G. Santos, and G. P. Guedes, "AffectPT-br: An affective Lexicon based on LIWC 2015," in *Proc. 37th Int. Conf. Chilean Comput. Sci. SoC. (SCCC)*, Nov. 2018, pp. 1–5.

[6] S. Hochreiter and J. J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Oct. 2014, pp. 1724–1734.

[8] U. Gupta, A. Chatterjee, and R. Srikanth, "A sentiment-and-semantics-based approach for emotion detection in textual conversations," *CoRR*, vol. abs/1707.06996, pp. 1–6, Jul. 2017.

[9] A. Sen, M. Sinha, S. Mannarswamy, and S. Roy, "Multi-task representation learning for enhanced emotion categorization in short text," in *Advances in Knowledge Discovery and Data Mining*. Springer-Verlag, Apr. 2017, pp. 324–336.

[10] P. Nandwani and R. Verma, "A review on sentiment analysis and emotion detection from text," *Social Netw. Anal. Mining*, vol. 11, no. 1, pp. 1–19, Dec. 2021.

[11] D. Yasmina, M. Hajar, and A. M. Hassan, "Using YouTube comments for text-based emotion recognition," *Proc. Comput. Sci.*, vol. 83, pp. 292–299, 2016.

[12] B. Desmet and V. Hoste, "Emotion detection in suicide notes," *Expert Syst. Appl.*, vol. 40, no. 16, pp. 6351–6358, Nov. 2013.

[13] M. Park, C. Cha, and M. Cha, "Depressive moods of users portrayed in Twitter," in *Proc. ACM SIGKDD Workshop Healthcare Informat. (HI-KDD)*, Jan. 2012, pp. 1–8.

[14] A. Joshi, V. Tripathi, R. Soni, P. Bhattacharyya, and M. Carman, "EmoGram: An open-source time sequence-based emotion tracker and its innovative applications," in *Proc. AAAI Workshops 13th AAAI Conf. Artif. Intell.*, Feb. 2016, pp. 512–516.

[15] Y. Wu, X. Kang, K. Matsumoto, M. Yoshida, K. Xielifuguli, and K. Kita, "Sentence emotion classification for intelligent robotics based on word Lexicon and emoticon emotions," in *Proc. IEEE Int. Conf. Intell. Robot. Control Eng. (IRCE)*, Aug. 2018, pp. 38–41.

[16] S. M. Mohammad and S. Kiritchenko, "Using hashtags to capture fine emotion categories from tweets," *Comput. Intell.*, vol. 31, no. 2, pp. 301–326, 2015.
[17] J. C. Platt, "Using analytic QP and sparseness to speed training of support vector machines," in *Proc. Conf. Adv. Neural Inf. Process. Syst. II*, vol. 11. Cambridge, MA, USA: MIT Press, 1999, pp. 557–563.
[18] A. Agrawal and A. An, "Unsupervised emotion detection from text using semantic and syntactic relations," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol.*, Dec. 2012, pp. 346–353.
[19] M. Hasan, E. Rundensteiner, and E. Agu, "EMOTEX: Detecting emotions in Twitter messages," in *Proc. ASE Bigdata/Socialcom/Cybersec. Conf.*, May 2014, pp. 27–31.
[20] M.-U. Asad, N. Afroz, L. Dey, R. P. D. Nath, and M. A. Azim, "Introducing active learning on text to emotion analyzer," in *Proc. 17th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2014, pp. 35–40.
[21] R. Johnson and T. Zhang, "Effective use of word order for text categorization with convolutional neural networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Dec. 2015, pp. 103–112.
[22] M.-H. Su, C.-H. Wu, K.-Y. Huang, and Q.-B. Hong, "LSTM-based text emotion recognition using semantic and emotional word vectors," in *Proc. 1st Asian Conf. Affect. Comput. Intell. Interact. (ACII Asia)*, May 2018, pp. 1–6.
[23] K. Tai, R. Socher, and C. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, Feb. 2015, pp. 1556–1566.
[24] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," Sep. 2014, *arXiv:1409.0473*.
[25] X. Li, S. Feng, D. Wang, and Y. Zhang, "Context-aware emotion cause analysis with multi-attention-based neural network," *Knowl.-Based Syst.*, vol. 174, pp. 205–218, Jun. 2019.
[26] C.-H. Wu, Z.-J. Chuang, and Y.-C. Lin, "Emotion recognition from text using semantic labels and separable mixture models," *ACM Trans. Asian Lang. Inf. Process.*, vol. 5, no. 2, pp. 165–183, Jun. 2006.
[27] I. Perikos and I. Hatzilygeroudis, "Recognizing emotions in text using ensemble of classifiers," *Eng. Appl. Artif. Intell.*, vol. 51, pp. 191–201, May 2016.
[28] L. I. Kuncheva, "Combining pattern classifiers: Methods and algorithms," *IEEE Trans. Neural Netw.*, vol. 18, no. 3, p. 964, May 2007.
[29] Z. Rajabi, A. Shehu, and O. Uzuner, "A multi-channel BiLSTM-CNN model for multilabel emotion classification of informal text," in *Proc. IEEE 14th Int. Conf. Semantic Comput. (ICSC)*, Feb. 2020, pp. 303–306.
[30] X. Sun, C. Zhang, and L. Li, "Dynamic emotion modelling and anomaly detection in conversation based on emotional transition tensor," *Inf. Fusion*, vol. 46, pp. 11–22, Mar. 2019.
[31] Z. Ahmad, R. Jindal, A. Ekbal, and P. Bhattachharyya, "Borrow from rich cousin: Transfer learning for emotion detection using cross lingual embedding," *Expert Syst. Appl.*, vol. 139, Jan. 2020, Art. no. 112851.
[32] M. Tang, B. Chen, X. Zhao, and L. Zhao, "Processing network emojis in Chinese sentence context: An ERP study," *Neurosci. Lett.*, vol. 722, Mar. 2020, Art. no. 134815.
[33] F. Jiang, Y. Liu, H. Luan, M. Zhang, and S. Ma, "Microblog sentiment analysis with emoticon space model," *J. Comput. Sci. Technol.*, vol. 30, no. 5, pp. 1120–1129, Sep. 2015.
[34] Y. Liu, D. Zhang, L. Du, Z. Gu, J. Qiu, and Q. Tan, "A simple but effective way to improve the performance of RNN-based encoder in neural machine translation task," in *Proc. IEEE 4th Int. Conf. Data Sci. Cyberspace (DSC)*, Jun. 2019, pp. 416–421.
[35] J. Cheng, P. Li, Z. Ding, S. Zhang, and H. Wang, "Sentiment classification of Chinese microblogging texts with global RNN," in *Proc. IEEE 1st Int. Conf. Data Sci. Cyberspace (DSC)*, Jun. 2016, pp. 653–657.
[36] R. Phan, T. M. Luu, R. Davey, and G. Chetty, "Biomedical named entity recognition based on hybrid multistage CNN-RNN learner," in *Proc. Int. Conf. Mach. Learn. Data Eng. (iCMLDE)*, Dec. 2018, pp. 128–135.
[37] G. A. Abandah, A. Graves, B. Al-Shagoor, A. Arabiyat, F. Jamour, and M. Al-Taee, "Automatic diacritization of Arabic text using recurrent neural networks," *Int. J. Document Anal. Recognit.*, vol. 18, no. 2, pp. 183–197, Jun. 2015.
[38] T. Liu, Y. Du, and Q. Zhou, "Text emotion recognition using GRU neural network with attention mechanism and emoticon emotions," in *Proc. 2nd Int. Conf. Robot., Intell. Control Artif. Intell.*, Oct. 2020, pp. 278–282.
[39] Q. Wang, L. Sun, and Z. Chen, "Sentiment analysis of reviews based on deep learning model," in *Proc. IEEE/ACIS 18th Int. Conf. Comput. Inf. Sci. (ICIS)*, Jun. 2019, pp. 258–261.
[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, U. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[41] L. L. Thomala. *Number of Monthly Active Users of Sina Weibo From 4th Quarter 2017 to 1st Quarter 2020(in Millions)*. Accessed: Sep. 9, 2020. [Online]. Available: https://www.statista.com/statistics/795303/china-mau-of-sina-weibo/
[42] T. Balomenos, A. Raouzaiou, S. Ioannou, A. I. Drosopoulos, and S. D. Kollias, "Emotion analysis in man-machine interaction systems," *Cogn. Emotion*, vol. 4, pp. 318–328, Jun. 2004.
[43] E. A. F. Biswas and S. Chadda, "Sentiment analysis with gated recurrent units," in *Proc. Adv. Comput. Sci. Inf. Technol. (ACSIT)*, 2015, vol. 2, no. 11, pp. 59–63.
[44] J. X. Chen, D. M. Jiang, and Y. N. Zhang, "A hierarchical bidirectional GRU model with attention for EEG-based emotion classification," *IEEE Access*, vol. 7, pp. 118530–118540, 2019.
[45] D. Li and J. Qian, "Text sentiment analysis based on long short-term memory," in *Proc. 1st IEEE Int. Conf. Comput. Commun. Internet (ICCCI)*, Oct. 2016, pp. 471–475.
[46] J. Wei, J. Liao, Z. Yang, S. Wang, and Q. Zhao, "BiLSTM with multi-polarity orthogonal attention for implicit sentiment analysis," *Neurocomputing*, vol. 383, pp. 165–173, Mar. 2020.
[47] M. Ling, Q. Chen, Q. Sun, and Y. Jia, "Hybrid neural network for Sina Weibo sentiment analysis," *IEEE Trans. Comput. Social Syst.*, vol. 7, no. 4, pp. 983–990, Aug. 2020.

CHANG LIU received the B.S. degree in software engineering from Zhengzhou University of Light Industry and the M.S. degree from Zhengzhou University. He is currently an Engineer in computer science with Chengdu Ruibei Yingte Information Technology Ltd. His research interests include big data and data mining, information retrieval, web mining, fault-tolerant computing, interconnection networks, and graph theory.

TAIAO LIU received the M.S. degree in computer science and technology from the School of Computer and Software Engineering, Xihua University, in 2021. He has published two papers on information retrieval: search engines, focused crawlers, and knowledge graphs. His experience and research work focus on social networks and software engineering.

SHUOJUE YANG is with the Department of Mechanical Engineering, School of Engineering, Johns Hopkins University. His experience and research work focus on big data, data analysis, and social networks.

YAJUN DU received the D.S. degree from the School of Computer and Communication, Southwest Jiaotong University, in 2005. He is currently a Professor of computer science at Xihua University (XHU). He has published several papers in the field of information retrieval and search engines. His experience and research work focus on information retrieval, search engines, web mining, and computer networks. He is serving on the committees of Chinese information and a PC Member for some international conferences, such as WISE, ICIC, CCIR, and WMSE.