IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 15, 2021, accepted February 5, 2021, date of publication March 2, 2021, date of current version March 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3062654

Refined Global Word Embeddings Based on Sentiment Concept for Sentiment Analysis

YABING WANG, GUIMIN HUANG, JUN LI, HUI LI, YA ZHOU, AND HUA JIANG

Guangxi Key Laboratory of Image and Graphic Intelligent Processing, Guilin University of Electronic Technology, Guilin 541004, China
School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, China

Corresponding author: Guimin Huang (sendhuang@126.com)

This work was supported by the National Natural Science Foundation of China under Grant 62066009 and Grant 61662012.

ABSTRACT Sentiment Analysis is an important research direction of natural language processing, and it is widely used in politics, news and other fields. Word embeddings play a significant role in sentiment analysis. The existing sentiment embeddings methods directly embed the sentiment lexicons into traditional word representation. This sentiment representation methods can only differentiate the sentiment information of different words, not the same word in different contexts, so it cannot provide accurate sentiment information for word in different contexts. This paper proposes sentiment concept to solve the problem. First, we found the optimal sentiment concept of words in Microsoft Concept Graph according to the context of words. Then we obtained the sentiment information of words under optimal sentiment concept from the multi-semantics sentiment intensity lexicon which we constructed in this paper to achieve accurate embedding of sentiment information and provide more accurate semantics and sentiment representation for words. Finally, we combined two refined word embeddings methods to achieve a more comprehensive word representation. Compared with traditional and sentiment embeddings methods on six representative datasets, the validity of the word embeddings method based on sentiment concept proposed in this paper is verified.

INDEX TERMS Deep learning, sentiment analysis, sentiment concept, word embeddings.

I. INTRODUCTION

Sentiment Analysis is a technology that automatically extracts sentiment information from unstructured texts. Sentiment Analysis is applied to many fields such as natural language processing (NLP), data mining and machine learning. Word vector representation is a key step in Sentiment Analysis. Nowadays, the widely used word embeddings technology is Word2Vec [1] and GloVe [2], which based on distributed representation. The idea is that words with similar contexts have similar vector representations. It is very useful for many tasks which related to semantic similarity because it can capture lots of contextual features to represent texts. However, it may produce opposite effect in Sentiment Analysis tasks. For example, ''cry'' and ''laugh'' have the same context in sentences ''she is crying'' and ''she is laughing'', so Word2Vec and GloVe will give ''cry'' and ''laugh'' similar vector representations. But analyzing in the view of sentiment, the sentiment polarity of ''cry'' and ''laugh'' are completely opposite. To solve this problem, researchers [3], [4] added sentiment information on the basis of Word2Vec and GloVe, which improved the accuracy of Sentiment Analysis. However, there are still have some problems in Sentiment Analysis. A word usually expresses different sentiments because it has different semantics in different contexts. For example, ''blue'' has different semantics in sentences ''He bought the blue hat'', ''He said it was just a blue'', and ''He is blue that nothing is going to get better'', which express different sentiments. The existing sentiment embeddings methods directly embed the sentiment lexicons into word representation, and there is no difference in the sentiment information of a word in different sentences, so it cannot provide precise sentiment information for word in different contexts to realize the accurate embedding. This paper proposes sentiment concept to solve the problem, we find out the optimal sentiment concept for the word according to the context to provide more accurate semantics and sentiment representation, and further improve the accuracy of Sentiment Analysis.

The main contributions of this paper are as follows: (1) proposing the sentiment concept to achieve the accurate embedding of sentiment information and provide

The associate editor coordinating the review of this manuscript and approving it for publication was Yu-Huei Cheng.

more precise semantics and sentiment representations for words, (2) constructing a sentiment intensity lexicon containing single-semantics and multi-semantics sentiment words through the multi-semantics integration of six representative sentiment intensity lexicons, to provide more accurate sentiment information for words with different semantics, (3) Refined-Word2Vec and Refined-GloVe we improved are averaged to obtain Refined Global Word Embeddings(RGWE). RGWE integrates not only different position features but also internal and external sentiment information. The validity of the word embeddings method based on sentiment concept proposed in this paper is verified through the experiment on six datasets with different categories and sizes.

The remainder of this paper as follows. Section II presents the related work of word embeddings in Sentiment Analysis. Section III detailed describes the word representation method RGWE proposed in this paper. Section IV contrasts and analyses the experimental results. Section V summarizes the work of this paper and looks forward to the future work.

II. RELATED WORK

With the development of NLP, Sentiment Analysis has been paid more attention by researchers and many efforts have been made in word embeddings. Jiang *et al.* [5] proposed Bag-of-words text representation method based on sentiment topic words, which is composed of deep neural network, sentiment topic words and context information, and performed well in Sentiment Analysis; Rezaeinia *et al.* [6] proposed refined word embeddings method based on Part-of-Speech(POS) tagging technology and sentiment lexicons, which improved the performance of pre-trained word embeddings in Sentiment Analysis; Pham *et al.* [7] proposed a joint model of multiple Convolutional Neural Networks (CNNs), which is focused on word embeddings from Word2Vec, GloVe and the one-hot character vectors, and it achieved good performance in aspect sentiment classification tasks; Zhou *et al.*[8] constructed a text representation model containing TF-IDF and topic features based on LDA for Sentiment Analysis, which reduced the dimension of word vector space in the traditional representation model; Han *et al.* [9] built a hybrid neural network model using convolutional neural networks and long short-term memory(LSTM) for document representation, and it incorporated user's and product's information; Devlin *et al.* [10] proposed the BERT model to represent text, which can better reflect the modifying relationship between words in texts, and it had good performance in Sentiment Analysis tasks; Liu *et al.* [11] proposed latent topic information of the text that used Neural topic model into word-level semantics representations to deal with the problem of data sparsity, and presented a new topic-word attention mechanism to explore the semantics of words from the perspective of topic word association; Li *et al.* [12] proposed a framework that combined different levels of prior knowledge into word embeddings for Sentiment Analysis, which improved the performance of Sentiment Analysis; Xu *et al.* [13] proposed an improved word representation method, which integrated the contribution of sentiment information into the traditional TF-IDF algorithm and generated weighted word vectors, and the method had higher F1 score; Peters *et al.* [14] proposed a text representation model based on deep learning framework, and it constructed an English text representation model which contained grammar feature, semantics feature and sentiment feature by training a large number of sentiment text corpus; Hao *et al.* [15] proposed a method for cross domain sentiment classification using random embeddings, which retained similar structure in embedding space and achieved well results in the task of Sentiment Analysis; Usama *et al.* [16] merged multilevel features which are from different layers of the same network and different network architectures to improve the accuracy of Sentiment Analysis; Majumder *et al.* [17] demonstrated the correlation between sarcasm detection and sentiment classification, and proposed a multi-tasking learning framework to improve the performance of two tasks; Ma *et al.* [18] proposed Sentic LSTM to explicitly integrate the explicit knowledge with implicit knowledge, and proposed an extension of Sentic LSTM to concern with a joint task combining the target-dependent aspect detection and targeted aspect-based polarity classification; Cambria *et al.* [19] used Common Sense Computing to enhance the capability of perceiving and expressing emotions of computers, and improved the human-computer interaction; Akhtar *et al.* [20] proposed a stacked ensemble method to predict sentiment intensity by using a multi-layer perceptron network, which combed the outputs with deep learning and classical feature-based models; Gu *et al.* [21] proposed a word vector refinement model to refine pretrained word vectors using sentiment intensity scores provided by sentiment lexicons, which improved each word vector and performed better in Sentiment Analysis.

The researchers added sentiment information of words in representation methods, but it still cannot achieve accurate embedding of sentiment information. In this paper, we provide more accurate semantics and sentiment representation for words by the proposed method.

III. PROPOSED METHOD

In this part, we will elaborate RGWE method which is proposed in this paper. First, we embed different features such as POS, position, sentiment and sentiment concept in the original word vectors, which generated by Word2Vec and Glove, to obtain Refined-Word2Vec and Refined-GloVe. Then we average the representation of Refined-Word2Vec and Refined-GloVe to obtain RGWE, which integrates not only different position features but also internal and external sentiment information.

A. WORD2VEC MODEL & GLOVE MODEL

Word2Vec is a widely used word embeddings model, which can obtain the distributed vector representation of words from large amounts of data. Word2Vec contains CBOW model that predicts words by context information, and skip-gram model that predicts context by word information. The two

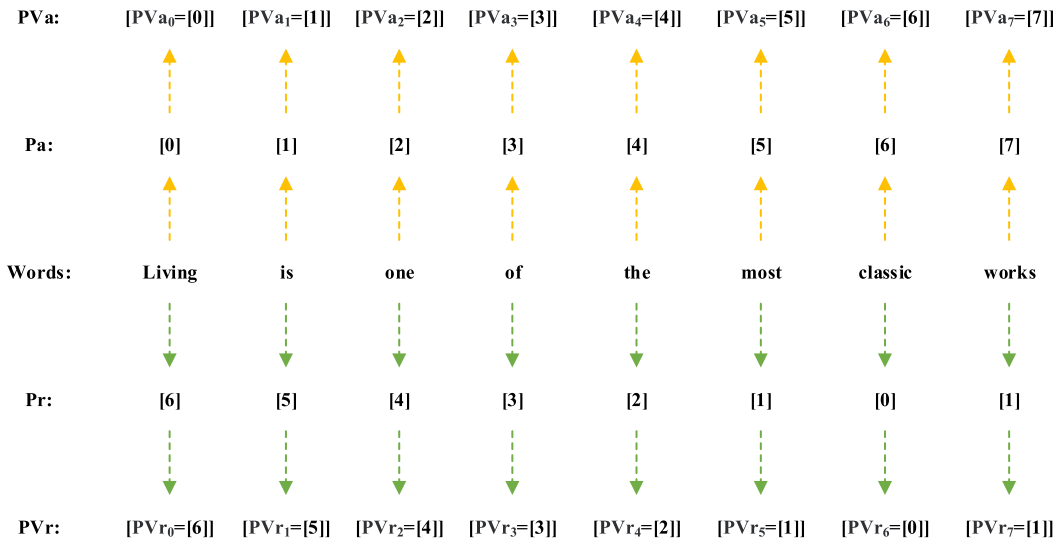


FIGURE 1. Position encoding of words.

models include input layer, projection layer and output layer, and both of them can provide accurate word embeddings representation. This paper adopts skip-gram model to represent text.

Glove is another popular word embeddings model, which get the word vector representation based on the global co-occurrence matrix. It proposes an attenuation function according to the distance between two words in the context window to calculate weight, so the farther apart the two words in contexts, the lower the weight.

B. POS

POS tags can provide syntactic information for words. In Sentiment Analysis, the POS information of words is very important for sentiment recognition. Different POS of words usually express different semantics and sentiments. For example, when the POS of ''novel'' is noun, its meaning is story and does not express sentiment, but when its POS is adjective, the meaning of ''novel'' is fancy and expresses positive sentiment. In this paper, the Stanford parser is used to tag the POS of words, and then convert the POS information into vectors and connect with word vectors of Word2Vec / GloVe. In this way, the Refined-Word2Vec / Refined-GloVe vectors will have POS features of words.

C. POSITION

Yu *et al.* [22] analyses the role of position feature in Sentiment Analysis tasks. We embed the absolute and relative position features of words in Refined-Word2Vec and Refined-GloVe respectively. The detailed description is as follows:

1) ABSOLUTE POSITION

We encode the absolute position of words and convert it to vector representation. As shown in Figure 1, the absolute position $Pa = [0, 1, 2, 3, 4, 5, 6, 7]$ of the sentence ''Living is one of the most classic works'' is converted to absolute position vector $PVa = [PVa_0, PVa_1, PVa_2, PVa_3, PVa_4, PVa_5, PVa_6, PVa_7]$, and finally connect the absolute position feature vector of words with the vector representation of Refined-Word2Vec.

D. RELATIVE POSITION

In Refined-GloVe, we use the equally important relative position feature and the absolute position feature in Refined-Word2Vec together to provide more comprehensive position information. About relative position feature, we consider that the word which closer to sentiment word contributes more for the sentiment judgment of sentences. For example, in the sentence ''Living is one of the most classic works'', the word ''most'' is closer to the sentiment word ''classic'', deeply reflects the sentiment degree of the sentence. In relative position coding of words, we set the position of sentiment word as 0, and the position of context words is the relative distance between sentiment word and them. As shown in Figure 1, the relative position $Pr = [6, 5, 4, 3, 2, 1, 0, 1]$ of the sentence ''Living is one of the most classic works'' is converted to relative position vector $PVr = [PVr_0, PVr_1, PVr_2, PVr_3, PVr_4, PVr_5, PVr_6, PVr_7]$ and finally connect the relative position feature vector of words with the vector of Refined-GloVe.

E. SENTIMENT LEXICONS

There are binary sentiment lexicons (such as Hu and Liu [23]), multi-classification sentiment lexicons (such as Riloff and Wiebe [24]), affective lexicons (such as Strapparava and Valitutti [25]), and sentiment intensity lexicons. In this paper, we choose sentiment intensity lexicons rather than sentiment polarity lexicons because the former can provide more detailed and comprehensive sentiment information for words. The detail of the sentiment intensity lexicons we selected and the Fusion Sentiment Intensity Lexicon (FSIL) we constructed is shown in Table 1.

TABLE 1. Details of sentiment intensity lexicons.

Refs.	Lexicon	Size	Scores Ranges
Nielsen FA. [27]	AFINN	2477	[-5, +5]
Taboada M. [28]	SO-CAL	6306	[-5, +5]
Mohammad S M *et al.* [29]	NRC Hashtag Sentiment Lexicon[1]	54129	[-7, +7]
Zhu X *et al.* [30]	NRC Emoticon Lexicon	62468	[-5, +5]
Cambria E *et al.* [31]	SenticNet 5	100000	[-1, 1]
Baccianella S *et al.* [32]	Sentiwordnet 3.0[2]	117659	[0, 1]
—	Fusion Sentiment Intensity Lexicon (FSIL)	172677	[-1, +1]

In the six sentiment intensity lexicons that we selected, sentiwordnet 3.0 does not directly gives sentiment score for each semantics of sentiment words, but gives positive and negative scores in the interval [0,1] for each semantics of sentiment words. We calculate the sentiment score for each semantics of sentiment words in sentiwordnet 3.0 using formula (1) [26]:

$$SentiScore = Pos_{score} - Neg_{score} \tag{1}$$

Pos_{score} is the positive score of sentiment word in a semantics, Neg_{score} is the negative score of sentiment word in that semantics, $SentiScore$ is the sentiment score of sentiment word in that semantics in sentiwordnet 3.0. Then we use the normalized method to map the sentiment scores of six sentiment intensity lexicons to the interval $[-1, +1]$.

We consider the sentiment information of sentiment word in different semantics. Therefore, we analyze the semantics and sentiment information of all sentiment words in the six lexicons as follows:

a. The sentiment word w exists in a sentiment lexicon with a semantics, then w exists in FSIL with the same semantics and sentiment score;
b. The sentiment word w exists in multiple sentiment lexicons with a semantics, we using formula (2) to calculate the sentiment score of w in FSIL:

$$SentiScore_w = \frac{\sum_{r=1}^{R} SentiScore_{w\_r}}{R} \tag{2}$$

$SentiScore_w$ is the sentiment score of w in FSIL, R is the number of sentiment lexicons which w is in, $SentiScore_{w\_r}$ is the sentiment score of w in the sentiment lexicon r;
c. The sentiment word w exists in one or more sentiment lexicons with multiple semantics. Firstly, we calculate the semantic similarity between different semantics of w with the cosine formula. Then set the semantic similarity threshold H. For the similar semantics group (SSG) whose semantic similarity greater than H, we select a semantics of SSG randomly as the semantics representation of SSG, and the average sentiment score of SSG as $SentiScore_w$ of the semantics in FSIL. For the semantics whose semantic similarity less than H (it is not similar to other semantics), it exists in FSIL as another semantics of w. Therefore, each sentiment word w in FSIL may correspond to one or more semantics and different sentiment information.

Through the integration, de-duplication, combination of semantics and calculation of sentiment score of six sentiment lexicons, we obtain FSIL from 343,039 sentiment words. FSIL contains 172,677 sentiment words, and among them 144,531 sentiment words have multiple semantics and 28146 sentiment words have single semantics.

F. SENTIMENT CONCEPT

We compare the words in the sentences with the sentiment words in FSIL to judge whether it is a sentiment word. Words in the sentences are context words except the sentiment words. A word can convey different sentiments depending on its context. This is because a word has multiple semantics and belongs to different sentiment concepts in different contexts respectively. For example, the sentiment concept of "pink" in sentence "I like pink skirts" is "color", which expresses neutral sentiment. Whereas its sentiment concept in sentence "He is the pink in the Foreign Office" is "elite", which expresses positive sentiment. Therefore, it is very important to determine the sentiment concept of words in the Sentiment Analysis tasks, which can determine the sentiment information in different contexts. The sentiment concept library used in this paper is Microsoft Concept Graph.[3] For the sentence $S = \{w_1, w_2, \ldots, w_m\}$, we predict the probability distribution of sentiment concept of the word w_i refer to formula (3) [33]:

$$p(c \mid V) = \frac{exp(c.V)}{\sum_{c_i \in C(w)} exp(c_i.V)} \tag{3}$$

$C(w)$ is the candidate concept set of w_i in Microsoft Concept Graph, V is the vector representation of S, which calculated using formula (4):

$$V = \frac{1}{m} \sum_{i=1}^{m} e_i \tag{4}$$

where e_i is the vector representation of w_i.

Then we choose the sentiment concept with the highest probability as the optimal sentiment concept of w_i:

$$c_{optimal} = arg\ max\ p(c \mid V) \tag{5}$$

After obtaining the optimal sentiment concept of words, we embed internal and external sentiment information under the optimal sentiment concept in Refined-Word2Vec and Refined-GloVe respectively.

1) INTERNAL SENTIMENT INFORMATION EMBEDDINGS

The process of embedding internal sentiment information in Refined-Word2Vec is shown in Figure 2. The detailed process is as follows:

[1] www.purl.com/net/sentimentoftweets
[2] http://sentiwordnet.isti.cnr.it
[3] https://concept.research.microsoft.com/

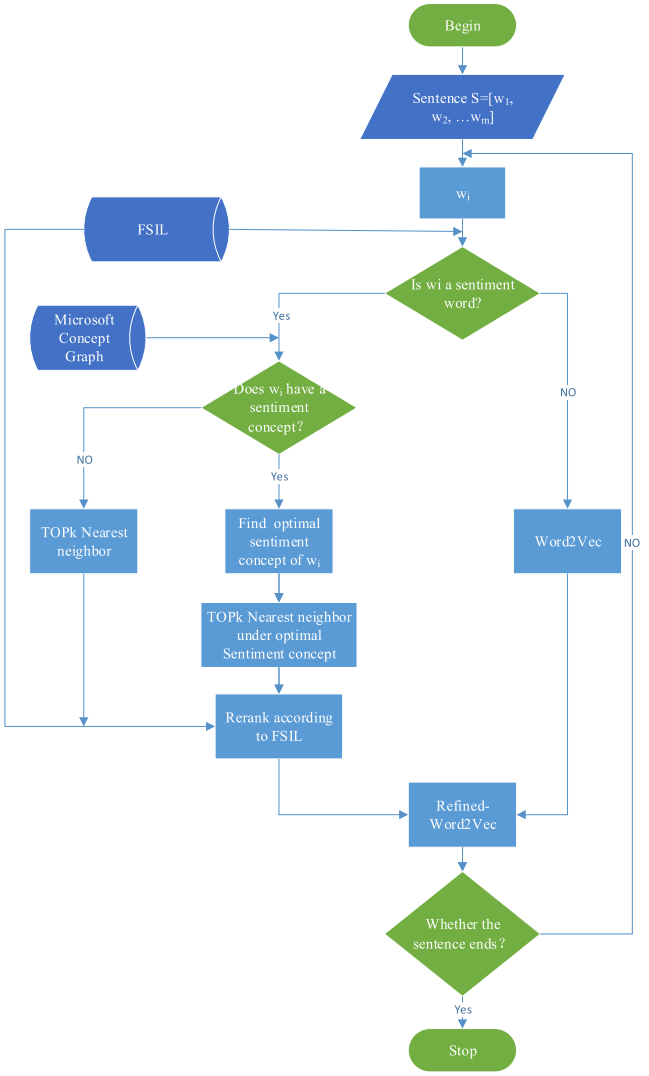


FIGURE 2. The flow chart of internal sentiment information embedding.

(1) Traverse the sentiment lexicon FSIL to judge whether w_i is a sentiment word or not;
(2) If w_i is a sentiment word, we find the optimal sentiment concept $c_{optimal}$ of w_i from Microsoft Concept Graph;
(3) Select TOP_k similar words with the highest semantic similarity under $c_{optimal}$ of w_i;
(4) Find the sentiment intensity score under $c_{optimal}$ of w_i and TOP_k in FSIL, and then reorder TOP_k according to the sentiment intensity difference between TOP_k and w_i. The smaller difference means more similar sentiments to w_i and the higher ranking. When determining the sentiment intensity score of TOP_k in FSIL, there are two situations:
   a. TOP_i is in FSIL. We find the sentiment intensity score under $c_{optimal}$ of TOP_i in FSIL;
   b. TOP_i is not in FSIL. We set the sentiment intensity score of TOP_i to 0.

In fact, TOP_i is rarely not in the FSIL because Microsoft Concept Graph aggregates similar words into a concept. Therefore, when a word is sentiment word, other words in the same concept are often sentiment words as well.

The objective function of embedding internal sentiment information in Refined-Word2Vec refers to formula (6) [21]:

$$argmin(V) = argmin \sum_{i=1}^{n} \left[\alpha dist\left(v_i^{t+1}, v_i^t\right) + \beta \sum_{j=1}^{10} w_{ij} dist\left(v_i^{t+1}, v_j^t\right) \right] \quad (6)$$

$$dist\left(v_i, v_j\right) = \sum_{d=1}^{D} \left(v_i^d - v_j^d\right)^2 \quad (7)$$

n is the number of target words that need to be refined. The first part represents the semantics vector distance between the refined vector representation v_i^{t+1} at step $t+1$ and v_i^t at step t of the target word w_i during the iterative optimization process. The second part represents the weighted sum of sentiment vector distance between v_i^{t+1} of w_i and v_j^t at step t of similar word w_j. We calculate the distance of D-dimensional vector v_i and v_j using formula (7). α and β are used to control the deviation degree of v_i^{t+1} and v_i^t and the closeness degree of v_i^{t+1} and v_j^t respectively. The sentiment contribution w_{ij} of w_j to v_i^{t+1} controls the movement direction of v_i^t, and α/β controls the movement distance of v_i^t.

In this paper, the sentiment contribution w_{ij} of w_j to v_i^{t+1} is calculated by using formula (8):

$$w_{ij} = \frac{1}{e^t} 0 \le t \ll 2 \quad (8)$$

t is the absolute difference between the sentiment intensity of w_i and w_j. The design of formula (8) comes from our thinking: the sentiment contribution w_{ij} of w_j to v_i^{t+1} decreases as the absolute difference t increases. Similar words w_j with smaller absolute difference from w_i contribute more to v_i^{t+1} than that with larger absolute difference. Formula (8) shows the difference in contribution of the sentiment information of similar words to the refined vector representation of target word more comprehensive.

2) EXTERNAL SENTIMENT INFORMATION EMBEDDINGS

we embed external sentiment information in Refined-Glove using formula (9). The first part contains the semantics information and sentiment information of words. The second part contains the sentiment concept information of words, which is used to restrict the semantics and sentiment range of words.

$$v_i = \gamma_i e_{ig} + \gamma e_c \quad (9)$$

e_{ig} is the original vector representation of w_i by GloVe, γ_i is the sentiment weight. e_c is the vector representation of the optimal sentiment concept $c_{optimal}$ of w_i, γ is the sentiment concept weight. Inspired by Xu *et al.* [13], the sentiment weight formula we designed is shown in formula (10) (11):

$$\gamma_i = TF - IDF_i * \tau \quad (10)$$

TF-IDF is the most commonly used method of weight calculation in text classification. We consider that the greater the sentiment intensity of words, the greater the

TABLE 2. Detailed statistics of the experimental datasets.

Dataset	Train	Valid	Test	Total	Classes	Balance
SemEval[10] (Nakov *etal.*,2013)	9684	1654	3813	15151	3 (positive/neutral/negative)	No
SST1[11](Socher *etal.*,2013)	8544	1101	2210	11855	5 (very positive/positive /neutral/negative/very negative)	No
SST2(Socher *etal.*,2013)	6920	872	1821	9613	2 (positive/negative)	No
IMDB[12](Pang *etal.*,2005)	40000	5000	5000	50000	2 (positive/negative)	Yes
Amazon[13](health)	8000	1000	1000	10000	2 (positive/negative)	Yes
Yelp 2014[14](Restaurant)	3072	384	384	3840	5 (very positive/positive /neutral/negative/very negative)	Yes

sentiment weight. We set different sentiment contribution value τ according to the sentiment intensity of words. We get the sentiment intensity under $c_{optimal}$ of words, and then calculate the corresponding sentiment contribution value according to formula (11):

$$\tau = \begin{cases} 1 & |SentiScore_w| = 0 \\ 6/5 & 0\,|SentiScore_w| < 0.2 \\ 7/5 & 0.2 \leq |SentiScore_w| < 0.4 \\ 8/5 & 0.4 \leq |SentiScore_w| < 0.6 \\ 9/5 & 0.6 \leq |SentiScore_w| < 0.8 \\ 2 & 0.8 \leq |SentiScore_w| \leq 1 \end{cases} \tag{11}$$

$SentiScore_w$ is the sentiment score of w in FSIL.

G. REFINED GLOBAL WORD EMBEDDINGS

After words are represented by two different vectors through Refined-Word2Vec and Refined-GloVe, respectively, We obtain RGWE by averaging the two different vectors representation, which comes from our consideration:

(1) The absolute position feature and relative position feature of words are embedded in Refined-Word2Vec and Refined-GloVe respectively, and RGWE integrates different position features to obtain more comprehensive position feature representation;

(2) The internal sentiment feature and external sentiment feature are embedded in Refined-Word2Vec and Refined-GloVe respectively, and RGWE integrates internal and external sentiment feature to obtain more comprehensive sentiment feature representation.

IV. EXPERIMENT

A. DATASETS

Six available classical public datasets are selected to evaluate the performance of RGWE proposed in Sentiment Analysis tasks. The details of datasets are shown in Table 2. For the datasets SemEval [34], SST1 [35] and SST2 [35] with standard train/valid/test, we experiment according to train/valid/test. For those without standard train/valid/test but completely balanced datasets IMDB [36], Amazon (health) [37] and Yelp 2014(Restaurant) [38], we experiment by stratifying sampling with 8:1:1 to obtain the corresponding train/valid/test. In addition, we set the intersection of training set and test set not to be empty to avoid the influence of technical terms on Sentiment Analysis.

B. EXPERIMENT SETTING

1) DATA PREPROCESSING

We perform general preprocessing for datasets: 1. Delete non-English words and special characters; 2. Delete stop words[9] and words with frequency less than 5; 3. Change all uppercase to lowercase; 4. Extend abbreviation[10] to ensure that sentiment words can be found in FSIL; 5. Stemming; 6. Text segmentation; 7. POS tagging.[11] We do not delete the short sentences because we consider that some short sentences also express sentiment, such as ''very good'', ''it is too bad'', etc.

2) WORD EMBEDDINGS METHODS

Word embeddings methods for comparison:

Traditional word embeddings: Word2Vec[12] (skip-gram) and GloVe[13];

Sentiment embeddings: SSWE[14];

Refined Embeddings: Seninfo+TF-IDF [13], Re(GLOVE) [21], Refined-Word2Vec, Refined-GloVe, and the RGWE that we proposed.

Word2Vec, GloVe, SSWE, Seninfo+TF-IDF, Re(GLOVE), Refined-Word2Vec, Refined-GloVe and RGWE are pre-trained on training datasets with 300-dimensions. We randomly assign word vectors for words that have not appeared in pre-trained.

3) DEEP LEARNING METHODS

We select three commonly used methods in deep learning to analyze sentiment texts:

Convolution Neural Networks (CNN): CNN captures the local feature information in texts by convolutional layer. The region sizes of convolution filter are (2,3,4), and 60 filters for each region size.

Bidirectional Long Short Term Memory Network (Bi-LSTM): Bi-LSTM captures the past and future contextual feature information of texts from forward and backward, and prevents the problems of gradient vanishing and gradient exploding. We adopt 2×128 hidden network units.

Bidirectional Gated recurrent units (Bi-GRU): Bi-GRU is a variant of Bi-LSTM, which reduces the gating on the basis

[9]http://nlp.stanford.edu/IR-book/html/htmledition/dropping-common terms-stop-words-1.html

[10]http://www.noslang.com/dictionary

[11]https://nlp.stanford.edu/software/tagger.shtml

[12] https://code.google.com/archive/p/Word2Vec/

[13]http://nlp.stanford.edu/projects/glove/

[14]http://ir.hit.edu.cn/~dytang/

TABLE 3. The F1-score of different embedding methods on datasets with CNN.

Method	Dataset	SemEval	SST1	SST2	IMDB	Amazon	Yelp 2014
Conventional Embeddings	Word2Vec [1]	62.3	45.7	84.1	84.5	84.4	42.1
	GloVe [2]	63.2	46.8	84.9	85.5	85.2	43.1
Sentiment Embeddings	SSWE	64.1	47.1	87.6	87.6	87.7	43.5
	Seninfo+TF-IDF [13]	66.7	49.1	88.8	89.0	89.0	45.4
Refined Embeddings	Re(GloVe) [21]	68.2	50.2	89.5	89.6	89.6	46.1
	Refined-Word2Vec	66.8	49.5	88.9	89.2	88.8	45.9
	Refined-GloVe	68.3	50.5	89.5	89.7	89.6	46.3
	RGWE	69.1	50.8	89.6	90.1	89.9	46.9

of Bi-LSTM, has simpler structure and fewer parameters. We adopt 2×128 hidden network units.

In addition, the datasets we selected are all short texts, the longest sentence has 326 words, and 97.8% of sentences are less than 110 words. Therefore, we set the maximum length to 110, sentences less than 110 words will be filled with 0 vector. We adopt dropout to prevent overfitting and set dropout to 0.5, and use tanh as the activation function of hidden layer and softmax as the classification function.

C. EXPERIMENT RESULTS

After verification of validation sets, we set the optimal k to 10, the optimal α:β= 0.03 for datasets SemEval, Amazon, Yelp 2014 and α:β= 0.1 for datasets SST1, SST2, IMDB respectively. We adopt F1-score which can comprehensively measure performance as the evaluation index. The experimental results are the average F1-score of running 10 times on test sets.

1) COMPARISON OF DIFFERENT WORD EMBEDDINGS METHODS

The comparison of experimental results among Refined-Word2Vec, Refined-GloVe, RGWE and Word2Vec, GloVe, SSWE, Seninfo+TF-IDF, Re(GLOVE) is shown in Table 3.

It can be seen from table 3 that the performance of sentiment embeddings methods SSWE and Seninfo+TF-IDF are better than traditional embeddings methods on datasets. This is because SSWE and Seninfo+TF-IDF contain sentiment information of words, it shows the importance of sentiment information on Sentiment Analysis. The improved Refined-Word2Vec and Refined-GloVe in this paper perform better than SSWE and Seninfo+TF-IDF. This is because, on the one hand, the sentiment intensity lexicon FSIL that contains 172677 sentiment words is embedded, which can provide more detailed sentiment feature; on the other hand, Refined-Word2Vec and Refined-GloVe contain sentiment features and sentiment concept features of words, which can capture the real sentiment of words in sentences more accurately. Refined- GloVe performs slightly better than Re(GloVe) on datasets SemEval, SST1 and Yelp 2014. Refined-GloVe and Re(GloVe) adopt different methods to embed the sentiment information, but Re(GloVe) lacks the sentiment concept information of words. The difference is more obvious on multi-classification datasets. The RGWE method has the best performance. The average F1 values are 89.86%, 69.1% and 48.85% for binary classification, ternary classification and fined-grained classification with CNN respectively. The reason is RGWE integrates not only different position features but also internal and external sentiment information.

2) COMPARISON OF DIFFERENT DEEP LEARNING METHODS

Table 4 shows the experimental results of different deep learning methods combined with RGWE on datasets. It can be seen that the combination of RGWE and Bi-GRU performs better than RGWE and Bi-LSTM. GRU is a variant of LSTM, and has fewer parameters than LSTM, so it is easier to converge. The performance of LSTM is better than GRU under large-scale datasets, while GRU has more advantages than LSTM under small-scale datasets. However, large-scale publicly sentiment analysis datasets are not easy to obtain. The scale of the classic and representative sentiment analysis datasets we choosing are not large enough, so GRU performs better than LSTM on our experimental data sets.

3) OPTIMAL SEMANTIC SIMILARITY THRESHOLD H

When constructing the sentiment intensity lexicon FSIL, if the semantic similarity threshold H is too large, similar semantics will appear in different SSGs, which causes it is difficult to determine the sentiment information of words in different SSGs; if the threshold H is too small, dissimilar semantics will appear in the same SSG, which cannot distinguish the sentiment information of different semantics in the same SSG. Therefore, we compare the performance of different threshold H on datasets to determine the optimal threshold H.

The performance of different threshold H on datasets is shown in Figure 3. It can be seen that the optimal threshold H is between 0.7 and 0.8. In this paper, we set H=0.78.

4) OPTIMAL SENTIMENT CONCEPT WEIGHT γ

Sentiment concept weight γ in Refined-GloVe is used to measure the contribution of sentiment concept to Sentiment Analysis. If the weight γ is too large, the contribution will be overestimated and reduce the accuracy of Sentiment Analysis; if the weight γ is too small, it cannot fully reflect the differences among various sentiment concepts. Therefore,

[10] http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=28685
[11] https://nlp.stanford.edu/sentiment/
[12] https://www.imdb.com/
[13] http://snap.stanford.edu/data/amazon-meta.html
[14] https://www.yelp.com/

TABLE 4. The F1-score of different deep learning methods on datasets with RGWE.

Method \ Dataset	SemEval	SST1	SST2	IMDB	Amazon	Yelp 2014
CNN+ RGWE	69.1	50.7	89.6	90.1	89.9	46.9
Bi-LSTM+ RGWE	70.0	51.6	90.3	91.0	90.6	47.3
Bi-GRU+ RGWE	70.8	52.3	91.2	91.3	91.1	47.9

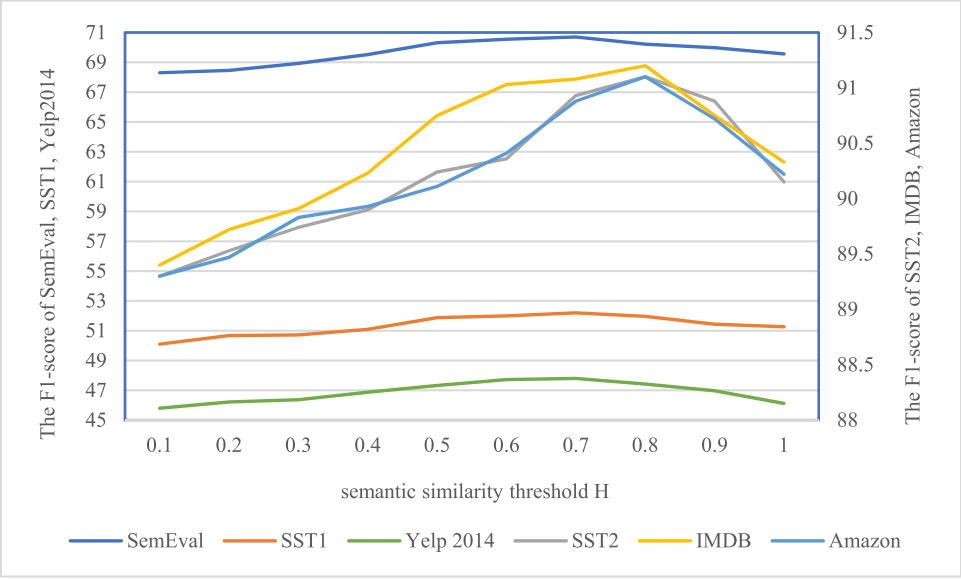


FIGURE 3. Performance comparison of different threshold H on datasets with Bi-GRU.

we compare the performance of different sentiment concept weight γ on datasets with H=0.78 to choose the optimal weight γ.

Figure 4 shows the performance of different values of weight γ in Refined-GloVe on datasets, which can be seen that the optimal sentiment concept weight is between 0.8 and 1.0. In this paper, we set $\gamma = 0.9$.

5) THE INFLUENCE OF SENTIMENT CONCEPT ON SENTIMENT ANALYSIS

We evaluate the influence of sentiment concept on Sentiment Analysis. In Table 5, Word2Vec+sen and GloVe+sen are the vector representation of adding features other than sentiment concept on the basis of Word2Vec and GloVe respectively. RGWE1 represents the average combination of Word2Vec and GloVe. RGWE1+sen represents the average combination of Word2Vec+sen and GloVe+sen.

It can be seen from Table 5 that the performance of embedding the sentiment feature and sentiment concept feature is better than that of purely sentiment lexicons. The F1-score of Refined-GloVe with sentiment concept exceeds GloVe+sen by 1.6%, 1.2%, 0.8%, 1.1%, 0.9% and 1.1% on six datasets respectively. The F1-score of RGWE with sentiment concept exceeds RGWE1+sen by 1.5%, 1.3%, 0.8%, 1.2%, 0.8%, and 1.6% on six datasets respectively, which shows the importance of sentiment concept on Sentiment Analysis. In addition, we find that sentiment concept has a greater impact on fined-grained classification than other classifications. This is because fined-grained sentiment classification is more detailed, and the expression of different classifications may be very similar (such as negative and very negative). Therefore, it is more important to distinguish the sentiment information of words in different semantics. In binary classification datasets, IMDB is more susceptible to sentiment concepts. We consider the reason is that IMDB has a larger amount of data and the sentiment expression in IMDB is more diverse.

6) THE INFLUENCE OF DIFFERENT TYPES SENTIMENT LEXICONS ON SENTIMENT ANALYSIS

Sentiment polarity lexicons and sentiment intensity lexicons both can be embedded in word representations to provide sentiment information for words. We compare the difference of embedding sentiment polarity lexicons and sentiment intensity lexicons. The sentiment polarity lexicon in the experiment is to set words with sentiment intensity greater than 0 in FSIL as positive words, and words with sentiment intensity less than 0 as negative words to get fusion sentiment polarity lexicon (FSPL).

As shown in Figure 5, the performance of embedding FSIL is better than FSPL, because FSIL provides more detailed sentiment information than FSPL for words, rather than simply distinguishing sentiment polarities.

7) THE INFLUENCE OF DIFFERENT SIZE SENTIMENT LEXICONS ON SENTIMENT ANALYSIS

The lexicon size from AFINN to FSIL is increasing in order. We compare the influence of lexicon size on Sentiment

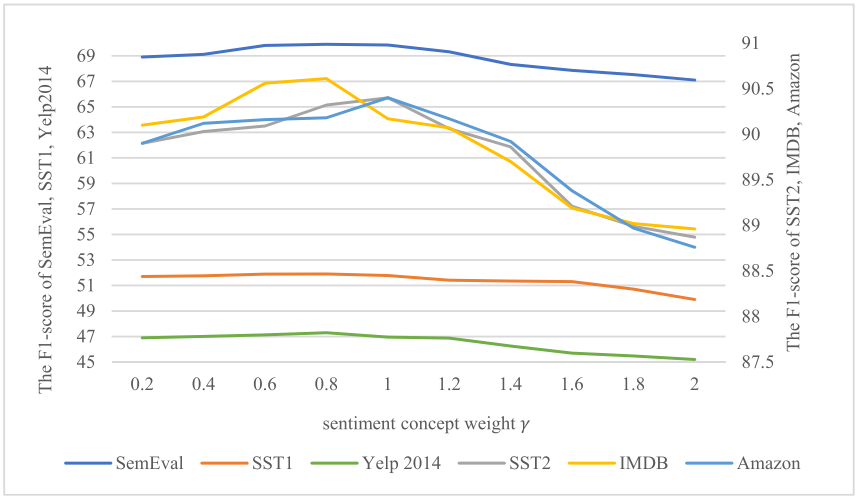


FIGURE 4. Performance comparison of different weight γ on datasets with Bi-GRU.

TABLE 5. The F1-score of sentiment concept on sentiment analysis with Bi-GRU.

Method \ Dataset	SemEval	SST1	SST2	IMDB	Amazon	Yelp 2014
Word2Vec	65.1	48.7	87.4	87.7	87.5	43.9
Word2Vec+sen	67.3	50.3	88.7	88.9	88.8	45.3
Refined-Word2Vec	68.9	51.6	89.6	90.1	89.7	46.5
GloVe	65.9	49.5	87.9	88.2	87.7	45.0
GloVe+sen	68.3	50.7	89.6	89.7	89.5	46.2
Refined-GloVe	69.9	51.9	90.4	90.8	90.4	47.3
RGWE1	65.4	49.2	87.6	88.5	87.3	44.3
RGWE1 +sen	69.3	51.0	90.4	90.1	90.3	46.3
RGWE	70.8	52.3	91.2	91.3	91.1	47.9

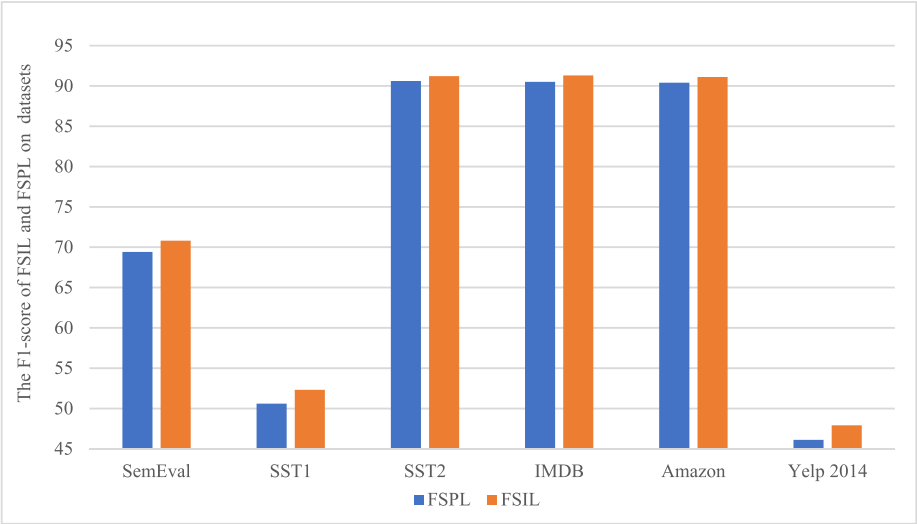


FIGURE 5. Performance of FSIL and FSPL on different datasets with Bi-GRU.

Analysis by embedding seven sentiment intensity lexicons with different sizes.

Figure 6 shows that the F1-score increases with the increase of sentiment lexicons size, which shows the influence of sentiment lexicons size on Sentiment Analysis. At the same time, it can be seen from figure 6 that the embedding of FSIL has the best performance on all datasets. This is because: (1) FSIL has the largest size (172677) among the seven sentiment lexicons; (2) 83.7% of sentiment words have multi-semantics and different sentiment intensities, which can provide more detailed sentiment information for words in different contexts.

D. ERROR ANALYSIS

In our experiment, there are some sentences with inaccurate analysis results. We conclude the following two reasons:

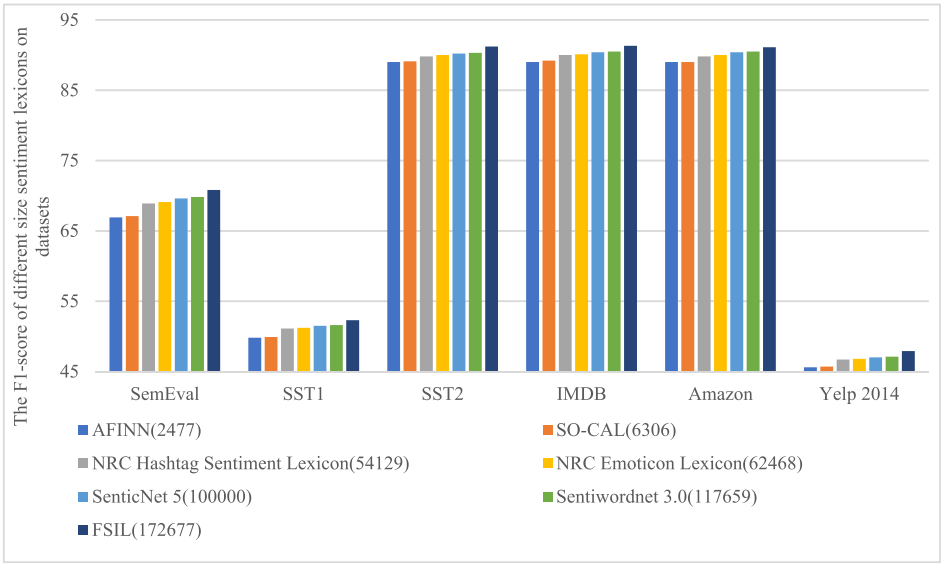


FIGURE 6. Performance of different size sentiment lexicons on datasets with Bi-GRU.

(1) The proportions of noun, verb, adjective and adverb are 63%, 6%, 21%, and 10% in FSIL, respectively. That is in FSIL, there are 63% of nouns sentiment words and 37% of non-noun sentiment words. While sentiment concepts can be found only for the noun sentiment words because the concepts and instances (words under concept) in Microsoft Concept Graph are nouns. This means that 37% of words in FSIL cannot be found the corresponding sentiment concept;

(2) It is inaccurate that there are 144531 multi-semantics and 28146 single-semantics sentiment words in FSIL. Single-semantics sentiment words may be multi-semantics because its other semantics not exist in FSIL or express neutral sentiment.

V. CONCLUSION

With the development of NLP technology, Sentiment Analysis has been applied in many fields, and the effect of sentiment analysis depends more on the quality of word embeddings, so it is necessary to study word embeddings methods in Sentiment Analysis tasks. This paper proposes the RGWE method based on sentiment concept to solve the problem that current word representation methods cannot accurately embed sentiment information in Sentiment Analysis tasks. We find the optimal sentiment concept of words according to the different contexts and provide more accurate semantics and sentiment representation for words. RGWE integrates not only different position features but also internal and external sentiment information by averaging Refined-Word2Vec and Refined-GloVe, which further improve the accuracy of Sentiment Analysis. The validity of RGWE is verified by comparing with the traditional embedding methods and sentiment embeddings methods on typical datasets. However, the concepts and the instances in Microsoft Concept Graph are only nouns, so the sentiment concept for verbs, adjectives, and adverbs are not be found in this paper. The problem will be studied in future work.

REFERENCES

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2013, pp. 1–12.

[2] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[3] D. Tang, F. Wei, B. Qin, N. Yang, T. Liu, and M. Zhou, "Sentiment embeddings with applications to sentiment analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 2, pp. 496–509, Feb. 2016.

[4] Y. Ren, Y. Zhang, M. Zhang, and D. Ji, "Improving Twitter sentiment classification using topic-enriched multi-prototype word embeddings," in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 3038–3044.

[5] Z. Jiang, S. Gao, and L. Chen, "Study on text representation method based on deep learning and topic information," *Computing*, vol. 102, no. 3, pp. 623–642, Sep. 2019.

[6] S. M. Rezaeinia, R. Rahmani, A. Ghodsi, and H. Veisi, "Sentiment analysis based on improved pre-trained word embeddings," *Expert Syst. Appl.*, vol. 117, pp. 139–147, Mar. 2019.

[7] D.-H. Pham and A.-C. Le, "Exploiting multiple word embeddings and one-hot character vectors for aspect-based sentiment analysis," *Int. J. Approx. Reasoning*, vol. 103, pp. 1–10, Dec. 2018.

[8] W. Zhou, H. Wang, and H. Sun, "A method of short text representation based on the feature probability embedded vector," *Sensors*, vol. 19, no. 17, pp. 185–209, Aug. 2019.

[9] H. Han, X. Bai, and P. Li, "Augmented sentiment representation by learning context information," *Neural Comput. Appl.*, vol. 31, no. 12, pp. 8475–8482, Dec. 2019.

[10] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2019, pp. 4171–4186.

[11] W. Liu, G. Cao, and J. Yin, "Bi-level attention model for sentiment analysis of short texts," *IEEE Access*, vol. 7, pp. 119813–119822, Sep. 2019.

[12] Y. Li, Q. Pan, T. Yang, S. Wang, J. Tang, and E. Cambria, "Learning word representations for sentiment analysis," *Cognit. Comput.*, vol. 9, no. 6, pp. 843–851, Dec. 2017.

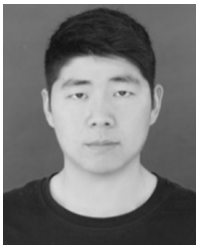
[13] G. Xu, Y. Meng, X. Qiu, Z. Yu, and X. Wu, "Sentiment analysis of comment texts based on BiLSTM," *IEEE Access*, vol. 7, pp. 51522–51532, Jan. 2019.

[14] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *J. Assoc. Comput. Linguistics*, vol. 1, pp. 2227–2237, Mar. 2018.

[15] Y. Hao, T. Mu, R. Hong, M. Wang, X. Liu, and J. Y. Goulermas, "Cross-domain sentiment encoding through stochastic word embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 10, pp. 1909–1922, Oct. 2020.
[16] M. Usama, W. Xiao, B. Ahmad, J. Wan, M. M. Hassan, and A. Alelaiwi, "Deep learning based weighted feature fusion approach for sentiment analysis," *IEEE Access*, vol. 7, pp. 140252–140260, Sep. 2019.
[17] N. Majumder, S. Poria, H. Peng, N. Chhaya, E. Cambria, A. Gelbukh, and E. Cambria, "Sentiment and sarcasm classification with multitask learning," *IEEE Intell. Syst.*, vol. 34, no. 3, pp. 38–43, May 2019.
[18] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain, "Sentic LSTM: A hybrid network for targeted aspect-based sentiment analysis," *Cognit. Comput.*, vol. 10, no. 4, pp. 639–650, Aug. 2018.
[19] E. Cambria, A. Hussain, C. Havasi and C. Eckl, "Sentic computing: Exploitation of common sense for the development of emotion-sensitive systems," in *Development of Multimodal Interfaces: Active Listening and Synchrony* (Lecture Notes in Computer Science), vol. 5967. Berlin, Germany: Springer, 2010, doi: 10.1007/978-3-642-12397-9_12.
[20] M. S. Akhtar, A. Ekbal, and E. Cambria, "How intense are you? Predicting intensities of emotions and sentiments using stacked ensemble [application notes]," *IEEE Comput. Intell. Mag.*, vol. 15, no. 1, pp. 64–75, Feb. 2020.
[21] S. Gu, L. Zhang, Y. Hou, and Y. Song, "A position-aware bidirectional attention network for aspect-level sentiment analysis," in *Proc. Int. Conf. Comput. Linguistics*, 2018, pp. 774–784.
[22] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Refining word embeddings using intensity scores for sentiment analysis," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 3, pp. 671–681, Mar. 2018.
[23] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 168–177.
[24] E. Riloff and J. Wiebe, "Learning extraction patterns for subjective expressions," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2003, pp. 105–112.
[25] C. Strapparava and A. Valitutti, "WordNet affect: An affective extension of WordNet," in *Proc. Conf. LREC*, 2004, pp. 1083–1086.
[26] F. H. Khan, U. Qamar, and S. Bashir, "A semi-supervised approach to sentiment analysis using revised sentiment strength based on SentiWordNet," *Knowl. Inf. Syst.*, vol. 51, no. 3, pp. 851–872, 2017.
[27] F. A. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis microblogs," in *Proc. ESWC Workshop Making Sense Microposts, Big Things Come Small Packages*, 2011, pp. 93–98.
[28] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Comput. Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.
[29] S. M. Mohammad, S. Kiritchenko, and X. Zhu, "NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets," in *Proc. 7th Int. Workshop Semantic Eval. Exercises (SemEval)*, Aug. 2013, pp. 116–122.
[30] X. Zhu, S. Kiritchenko, and S. Mohammad, "NRC-Canada-2014: Recent improvements in the sentiment analysis of tweets," in *Proc. Int. Conf. Comput. Linguistics*, 2014, pp. 443–447.
[31] E. Cambria, D. Hazarika, K. Kwok, and S. Poria, "SenticNet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings," in *Proc. Nat. Conf. Artif. Intell.*, 2018, pp. 1795–1802.
[32] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining," in *Proc. Lang. Resour. Eval.*, 2010, pp. 17–23.
[33] J. Cheng, Z. Wang, J.-R. Wen, J. Yan, and Z. Chen, "Contextual text understanding in distributional semantic space," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2015, pp. 133–142.
[34] P. Nakov, S. Rosenthal, Z. Kozareva, V. Stoyanov, A. Ritter, and T. Wilson, "Semeval-2013 task 2: Sentiment analysis in Twitter," in *Proc. SemEval*, 2013, pp. 312–320.
[35] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2013, pp. 1631–1642.
[36] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proc. 43rd Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2005, pp. 115–124.
[37] K. Baktha and B. K. Tripathy, "Investigation of recurrent neural networks in the field of sentiment analysis," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Apr. 2017, pp. 2047–2050.
[38] S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad, "NRC-Canada-2014: Detecting aspects and sentiment in customer reviews," in *Proc. 8th Int. Workshop Semantic Eval. (SemEval)*, 2014, pp. 437–442.

YABING WANG was born in Henan, China. She is currently pursuing the Ph.D. degree with the School of Computer Science and Information Security, Guilin University of Electronic Technology, China. Her current research interests include natural language processing and sentiment analysis.

GUIMIN HUANG received the Ph.D. degree from the School of Computer Science, East China University of Science and Technology, in 2005. He is currently a Full Professor with the Guilin University of Electronic Technology, China. His research interests include natural language processing and text mining.



JUN LI was born in Xinyang, China. He is currently pursuing the Ph.D. degree with the School of Computer Science and Information Security, Guilin University of Electronic Technology, China. His current research interests include natural language processing and text understanding.

HUI LI was born in Inner Mongolia, China. He is currently pursuing the Ph.D. degree with the School of Computer Science and Information Security, Guilin University of Electronic Technology, China. His current research interests include natural language processing and text understanding.

YA ZHOU received the master's degree from the School of Computer Science, Fudan University, China. She is currently a Full Professor with the Guilin University of Electronic Technology, China. Her research interests include distributed computing and data mining.

HUA JIANG received the Ph.D. degree from the China University of Geosciences, in 2006. He is currently a Full Professor with the Guilin University of Electronic Technology, China. His research interests include database systems and text mining.