applied sciences

MDPI

Article

Sentiment-Aware Word Embedding for Emotion Classification

Xingliang Mao [1,†,‡], Shuai Chang [2,‡], Jinjing Shi [2,*], Fangfang Li [2,*] and Ronghua Shi [2]

[1] Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China; gisor@163.com

[2] School of Computer Science and Engineering, Central South University, Changsha 410073, China; 164712054@csu.edu.cn (S.C.); shirh@csu.edu.cn (R.S.)

* Correspondence: 214009@csu.edu.cn (J.S.); lifangfang@csu.edu.cn (F.L.)

† Current address: National University of Defense Technology, Changsha 410073, China.

‡ These authors contributed equally to this work.

Received: 12 February 2019; Accepted: 22 March 2019; Published: 29 March 2019

Abstract: Word embeddings are effective intermediate representations for capturing semantic regularities between words in natural language processing (NLP) tasks. We propose sentiment-aware word embedding for emotional classification, which consists of integrating sentiment evidence within the emotional embedding component of a term vector. We take advantage of the multiple types of emotional knowledge, just as the existing emotional lexicon, to build emotional word vectors to represent emotional information. Then the emotional word vector is combined with the traditional word embedding to construct the hybrid representation, which contains semantic and emotional information as the inputs of the emotion classification experiments. Our method maintains the interpretability of word embeddings, and leverages external emotional information in addition to input text sequences. Extensive results on several machine learning models show that the proposed methods can improve the accuracy of emotion classification tasks.

Keywords: emotion classification; sentiment lexicon; text feature representation; hybrid vectorization; sentiment-aware word embedding

1. Introduction

With the rapid increase in the popularity of social media applications, such as Twitter, a larger amount of sentiment data is being generated. Emotional analysis has attracted much attention. At the same time, sentiment analysis for Chinese social network data has been gradually developed. In 2013, the second CCF (China Computer Federation) International Conference on Natural Language Processing and Chinese Computing (NLPCC) established the task of evaluating the emotions of Weibo, which attracted many researchers and institutions. The conference drove the development of emotional analysis in China. Weibo sites have been the main communication tool. They provide information that is more up-to-date than conventional news sources, and this has encouraged researchers to analyze emotional information from this data source. There are many differences between Weibo text and traditional long text, such as movie reviews in sentiment analysis. Firstly, they are short with no more than 140 characters. Secondly, words used in Weibo are more casual than those in official texts, and they contain a lot of noise, such as informal text snippets. For example, there are web-popular words, like "LanShouXiangGu" (a network language/buzzword, means feel awful and want to cry). Web-popular words might be seen as traditional words but represent different meanings or emotions. Finally, Chinese is largely different from English; it has more complex syntaxes and sentence structures. This increases the difficulty of emotional analysis in Chinese.

Text consists of many ordered words in the emotional analysis task. In order to process a text document mathematically, the text document is projected into the vector space. A text document is represented by a vector in the same vector space so that the document can be classified by a model. The BoW (bag-of-words) model is widely used in text processing applications. It is an approach to modeling texts numerically [1]. It processes texts regardless of word order and semantic structure and disregards context which means that it is unable to sufficiently capture complex linguistic features. At the same time, one of the drawbacks of BoW is its high number of dimensions and excessive sparsity. The appearance of word embedding overcomes this shortcoming.

Recently, word embedding based approaches [2,3] have learned from low-dimensional, continuously-valued vector representations using unsupervised methods over the large corpus. State-of-the-art word embedding algorithms include the C and W model [4], the continuous bag-of-words (CBOW) model, the Skip-Gram Word2Vec model [3], and the GloVe (Global Vectors for Word Representation) model [5]. Word embedding techniques have been shown to facilitate a variety of NLP (natural language processing) tasks, including machine translation [6], word analogy [7], POS (part of speech) tagging [8], sequence labeling [4], named entity recognition [9], text classification [10], speech processing [11], and so on. The principle behind these word embedding approaches is the distributional hypothesis that “You shall know a word by the company it keeps” [12]. By leveraging statistical information, such as word co-occurrence frequencies, the method could explicitly encode many linguistic regularities and patterns into vectors. It produces a vector space in which each unique word in the corpus is assigned a corresponding vector in the space, and words with similar contexts in the training corpus are located in close proximity. The feasibility of distribution assumptions has been confirmed in many experiments [13,14].

However, most existing word embedding algorithms only consider statistical information from documents [3,5,7]. The representations learned from these algorithms are not the most effective for emotional analysis tasks. In order to improve the performance of word embedding in emotional analysis tasks, a method that combines the traditional word embedding and provides prior knowledge from external sources is proposed. The knowledge of the polarity and intensity of emotional words can be obtained via public sentiment lexicons, and this sentiment information is not directly obtained in word co-occurrence frequencies, which can greatly enhance the performance of word embedding for emotional analysis. For example, “happy” and “sad” might appear in the same or a similar emotional context but represent different emotions, so it is not enough to learn the emotional information of these two words by counting the word co-occurrence. The method proposed in this paper builds sentiment-aware word embedding by incorporating prior sentiment knowledge into the embedding process.

Our primary contribution is therefore to propose such a solution by making use of the external emotional information, and propose the sentiment-aware word embedding to improve emotional analysis. While there is an abundant literature in the NLP community on word embedding for text representations, much less work has been devoted in comparison to hybrid representation (combining the diverse vector representations into a single representation).The proposed sentiment-aware word embedding is implemented by jointly embedding the word and prior emotional knowledge in the same latent space. The method tests on the NLPCC dataset label Weibo data with seven emotions. First, our method encodes the semantic relationships between words by traditional word embedding. Second, the method incorporates sentiment information of words into emotional embedding. Various combinations of word representations are used in this experiment. It is the hybrid sentiment-aware word embedding that can encode both semantics and sentiments of words. In the experiments, the results show that the two kinds of semantic evidence can complement each other to improve the accuracy of identifying the correct emotion.

The paper is organized as follows. Section 2 briefly introduces the emotion analysis method. Section 3 presents the details of the proposed methodology. Section 4 discusses the experimental arrangement. Section 5 summarizes the contents of the full text and discusses the future direction of development.

2. Related Work

The study of emotional analysis is roughly summed up into three categories: Rule-based analysis, unsupervised classification, and supervised classification.

Rule-based analysis is mainly performed together with the emotion lexicon. In English text analysis, Kamps proposed a distance measurement method to determine the semantic polarity of adjectives based on the synonym graph theory model. In the analysis of Chinese text, Zhu et al. introduced a simple method based on HowNet Chinese lexicon to determine the semantic direction of Chinese words [15]. Pan et al. identified six kinds of emotion expressed by Weibo with the lexicon-based method [16]. However, the lexicon-based method is low in accuracy, and the classifying quality is easily limited by the lexicon [17]. In particular, the lexicon-based method ignores contextual information.

Unsupervised classification analysis does not use tagged documents but relies on a documentation's statistical properties, NLP processes, and existing vocabulary, which has an emotional or polarizing tendency. Turney presented a simple, unsupervised learning algorithm for classifying reviews [18]. Lin and He proposed a novel, probabilistic modeling framework based on latent Dirichlet allocation (LDA), called the joint sentiment/topic model (JST), which detects sentiments and topics simultaneously from the text [19]. They also explored various ways to obtain prior information to improve the accuracy of emotional detection. Yili Wang and Hee Yong Youn [20] proposed a novel feature weighting approach for the sentiment analysis of Twitter data and a fine-grained feature clustering strategy to maximize the accuracy of the analysis.

The analysis based on supervised classification generates an emotion classification model with labeled training data. The effectiveness of supervised technologies depends on the features used in the classification task. The bag-of-words features and their weighting scheme are widely used in natural language processing, which provides a simplified representation of documents through various features. However, these methods have limitations in the task of emotional analysis. Word embedding drives the development of many NLP tasks through the low-dimensional continuous vector representations of words.

In the framework of the word embedding model, the word vector is generated according to the distribution hypothesis [12]. It has been found that learning vectors can clearly encode many linguistic regularities and patterns [21]. However, it is still not enough to rely solely on word-level distribution information collected from the text corpus to learn high-quality representations [22,23]. Auxiliary information has been shown to help learn task-specific word embedding to improve the performance in the tasks [24,25].

In order to improve the representation of word embedding, some research work has been proposed to incorporate various additional resources into the learning framework of word representation. Some knowledge-enhancing word embedding models incorporate lexical knowledge resources into the training process of word embedding models [22,23,26,27]. In the study of Levy and Goldberg [28], the grammatical context of the automatically generated dependency analysis tree was used for word representation training. Meanwhile, some people learn cross-language word embedding with a multilingual parallel corpus [29–31]. Fuji Ren and Jiawen Deng pointed out that background knowledge is composed of keywords and co-occurring words that are extracted from the external corpus and proposed a background knowledge-based multi-stream neural network [32]. Many effective analysis algorithms take advantage of existing knowledge to improve classification performance [33–35]. Wang et al. [33] studied the problem of understanding human sentiments from the large-scale collection of internet images based on both image features and contextual social network information and proved that both visual feature-based and text-based sentiment analysis approaches can learn high-quality models. Tang et al. proposed a learning sentiment-specific word embedding approach dubbed sentiment embedding, which retains the effectiveness of word contexts and exploits the sentiment of text to learn more powerful, continuous word representations [36].

3. Methodology

Constructing an effective features vector to represent text is a basic component in the NLP tasks. In view of the specific emotion classification task, we propose the sentiment-aware word embedding based on the construction of a hybrid word vector method containing emotional information.

This study introduces all of the details about constructing the hybrid feature representations in the following sections (as shown in Figure 1).

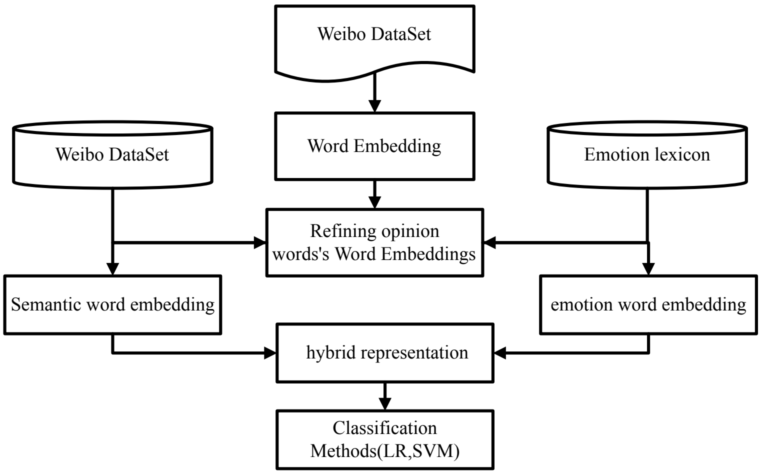

Figure 1. Framework for the proposed method.

The method comprises three main components: (1) the construction of semantic word vectors based on Word2Vec; (2) the construction of emotional word vectors based on the emotional lexicon; and (3) the construction of hybrid sentiment-aware word representations. The third step is based on the results of the previous two steps.

3.1. Constructing a Semantic Word Vector Based on Word2Vec

Word2Vec is an open source tool based on deep learning. It has become more and more popular recently because of its high accuracy in analyzing semantic similarities between two words and the relatively low computational cost. It has two modes: Continuous bag of words (CBOW) and Skip-Gram, which can be applied to quickly learn word embedding from the original text and capture word relationships with the built vector representation model (neural network model). In this paper, the Skip-Gram model was selected for word embedding training. After the text preprocessing step, the word vector representations of all words in the document are learned by the Skip-Gram model, and the vector representation for each word encountered in the input is extracted from the model. Then, the representation of each sentence can be received by averaging over the vectors of all its comprising words. The vector representation of the new document is derived in the same way during the sentiment prediction phase.

3.2. Constructed Emotional Word Vector Based on the Emotional Lexicon

After the preprocessing step, every word in each sentence is converted into a vector. The obtained vectors are then compared based on their cosine similarity degree with the vectors of the emotional words. The emotional word vectors with high similarity scores are selected to be combined. The proposal helps to increase the "semantic power" of traditional semantic space models for emotional analysis by combining different sources of semantic evidence. The hybrid sentiment-aware word embedding is inputted as the experimental model, showing that two kinds of semantic evidence can complement each other, and the mix of them can identify the correct emotion.

The lexicon-based feature extraction method is promising due to the existence of the emotional lexicon. Typically, an emotional lexicon consists of a set of language-specific words, which include information about the emotional category to which it belongs, the polarity intensity, and so on. A fine-grained multi-emotional polarity lexicon can improve the classification accuracy in emotional analysis tasks compared to simple emotional dictionaries [37].

A set of documents and emotional lexicons is given to build the model and generate the emotional word vectors. A corpus, D, consists of a set of texts, $D = \{d_1, d_2 \cdots, d_n\}$, and the vocabulary, $T = \{t_1, t_2 \cdots, t_m\}$, which are unique terms extracted from D. The word representation of the terms t_i are mapped from the Word2Vec model, and then a set of word representations of all words in the vocabulary is derived, i.e., $V = \{v_1, v_2 \cdots, v_m\} \in \mathbb{R}^{m \times d}$, where m is the size of the vocabulary, and d represents the dimensions of the vector space.

The core of this method is to construct emotional word vectors representing the emotional information for each word in the vocabulary, but most of them express some emotions that are not typical opinion words, so the paper introduces the lexicon-based method. In order to construct the emotional word vectors, the method utilizes all emotional words in the lexicon as the emotional vocabulary, $E = [E_1, E_2, \ldots. E_k]$, and gets their word representations, $V_E = \left[V_{E_1}, V_{E_2}, \cdots, V_{E_k}\right]$. Due to the scale problem of word embedding the model by training the corpus, there is a low degree of the coverage problem, that is, there are some words that are in the emotional vocabulary but not in the vector spaces learned from Word2Vec. Thus, these words can be ignored and deleted in subsequent processing.

The word embedding model captures the semantic relationships in the corpus text, and the contexts of words with the same emotional polarity are identical. Therefore, the similarity between two items is estimated by calculating the cosine similarity between the vectors represented by V and E, that is, $\text{sim}\,(\text{V}, \text{E}) = consine\,(\text{V}, \text{E})$, defined as

$$\text{sim}\,(\text{V}, \text{E}) = \frac{\sum_{i=1}^{d} \left(V_i * V_{E_i}\right)}{\sqrt{\sum_{i=1}^{d} V_i^2 * \sum_{i=1}^{d} V_{E_i}^2}} \tag{1}$$

where V and E are the vectors of length d.

For each item in T, the similarity to all items in E is calculated and then the similarity results are collected into the matrix $Y \in \mathbb{R}^{m \times k}$, where m is the length of the text glossary, k is the length of the emotional vocabulary, and $Y_{i,j}$ indicates the cosine similarity between the lexical item i and the emotion j. Based on matrix Y, the top n emotional words, $E_w = \{e_1, e_2, \cdots e_n\}$, are selected by setting the threshold. For words in E_w, they are the nearest neighbors to the item in T as determined by distinguishing their sentimental differences.

Similarity in Y means that an item in E and an item in T have the same context; meanwhile, the emotional intensity provided in the dictionary represents the emotional information of the item in E and constructs the emotion vector for all items in T by combining the two kinds of information.

The emotional word vector for each word is calculated as a weighted sum rather than a simple average operation:

$$EmotionVec_i = \frac{1}{n} \sum_{i=1}^{n} weight_i * V_{e_i}. \tag{2}$$

Then, Formula (3) is used to compute the weight of every word; based on this formula, the higher ranked nearest neighbors will receive higher weights:

$$weight_i = Y_{ij} * score_j. \tag{3}$$

The Dalian University of Technology Information Retrieval (DUTIR) is used to represent sentiment lexicon in this study, which is a Chinese ontology resource labeled by the Dalian University of Technology Information Retrieval Laboratory [38]. It contains 27,466 words, and each word is associated with a real-valued score of 1, 3, 5, 7, or 9. The score represents the degree of sentiment, where 9 indicates the maximum strength, and 1 is the minimum strength. Then, we rank the emotional words in E_w by strength scores. The $score_j$ is defined as the reciprocal rank of e_j in the E_w, that is

$$score_j = \frac{1}{rank_j} \tag{4}$$

where $rank_j$ denotes the rank of e_j generated by the intensity-based ranking process.

The weights have to be normalized to sum to one. This can be obtained with the following relation:

$$weight_i = \frac{weight_i}{\sum_{j=1}^{n} weight_j}. \tag{5}$$

In the process of constructing the emotional word vector of words, the shortcomings of the context-based word embedding result in words with opposite sentiment polarity having a fairly high cosine similarity in the vector space. There are great errors generated in the construction of emotional word vectors using word embedding mapping and cosine similarity, so sentimental lexicons are used to correct the word representations mapping from the word embedding space. It is a good way to optimize existing word vectors by using real-valued sentiment intensity scores provided by the emotional lexicons and word vector refinement model. In this way, the words are closer to semantically and emotionally similar words in the dictionary (that is, those with similar intensity scores) and stay away from words that are not emotionally similar but are similar in semantics.

3.3. Constructing Hybrid Sentiment-Aware Word Embedding

In this work, sentiment-aware word embedding is constructed to represent each word by connecting the emotional word vectors based on the lexicon and the semantic word vectors based on the Word2Vec model. Sentiment-aware word embedding can capture the emotional orientation of words, which is the word representation method strictly based on word embedding. In addition, it also makes the most of the contextual and semantic expression ability of the word embedding model.

The sentiment-aware word embedding combines the emotional word vectors with semantic word embedding to simplify combinatorial functions, which indicates that the advantages of the two models can be combined in a single mixed representation. This paper explores different methods of vector combination and experiments with the proposed vector combination method in Section 4. In particular, in order to compare the advantages and limitations of various methods, a comparative study of the two combination methods is conducted.

The first method combines the emotional word vectors with semantic words of a given word directly, which allows two vector representations with different dimension representations:

$$x_{new} = x_e \frown x_s \tag{6}$$

where x_e (x_s) represents the emotional word vector, (semantic word embedding).

Two vectors, x_c and y_e, form by linking the corresponding vectors from the original space x_1 with x_2, y_1, and y_2 as well. Cosine similarity is used to estimate the similarity between two items, and the key factor for cosine similarity is the dot product, i.e.,

$$x_c \cdot y_c = (x_1 \frown x_2) \cdot (y_1 \frown y_2) = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = x_1 \cdot y_1 + x_2 \cdot y_2. \tag{7}$$

Thus, the cosine similarity in cascade space is determined by the linear combination of the dot products of the vector component. Therefore, the semantic relations and emotional relations between two words are distinguished as features.

The second method is to combine these representational spaces by addition, which requires the two spaces to have the same dimensions, and it can be realized using simple vector addition. The value for each word in the new space is the normalized sum of its two component spaces, i.e.,

$$x_{new} = x_e + x_s. \tag{8}$$

From the dot product result of the vector, the direct superposition of the two vectors which combine the characteristic components of them increases the distinction between different emotional features.

No matter whether vector connection or vector overlay is used, the experimental results show that the advantages of the two models can be combined in a single mixed representation by combining word vector components generated in different ways with simple combinatorial functions.

3.4. Algorithm Implementation

According to data pre-processing, word embedding construction, emotional lexicon processing, and word similarity computation, Algorithm 1 presents the sentiment-aware word embedding approach to emotional analysis. From the algorithm description, it can be seen that with the first iteration, we can get the vocabulary. With the second and third iterations, we can get the vector of each item in the vocabulary and emotional lexicon. Based on the previous step, we can get the hybrid sentiment-aware word embedding through the last iteration.

Algorithm 1 sentiment-aware word embedding for emotion classification

```
Input: a set of unclassified text documents
Output: a set of classified text documents
for each doc ∈ corpus do
    for each term ∈ doc do
        terms ← pre-processing all term with NLP methods
    end
    vocabulary ← terms
end
for each term ∈ vocabulary do
    $SemanticVector_{term} \leftarrow w2vec_{term}$
end
for each emotional_word ∈ emotional lexicon do
    $Vector_{emotional\_word} \leftarrow w2vec_{emotional\_word}$
end
for each term ∈ vocabulary do
    for each emotional_word ∈ emotional lexicon do
        $sim(term, emotional\_word) = consine(term, emotional\_word)$
        if $sim(term, emotional\_word) > Threadhold$ then
            $weight = sim(term, emotional\_word) * score(emotional\_word)$
        end
    end
    $EmotionVector_{term} = \sum_{\text{emotional\_word}} \text{weigh}$
    $HybridVector1_{term}$=$SemanticVector_{term} + EmotionVector_{term}$
    $HybridVector2_{term} = SemanticVector_{term} \frown EmotionVector_{term}$
end
```

4. Experiments

In this section, the proposed hybrid word vector representation method is evaluated with a series of experiments on Weibo text datasets. Based on the above steps, the hybrid word vector is constructed and used as an input to train classification models. The experiments select different approaches to evaluate the proposed hybrid feature vectors. Then, a concrete discussion is given based on different classification results.

The Word2Vec model utilizes the genism software package in Python to create the word vectors, and the Skip-Gram, one method of Word2Vec, was selected for this work. Word2Vec works in a way that is comparable to the machine learning approach of deep neural networks, which allows the representation of semantically similar words in the same vector space with adjacent points.

4.1. Description of Datasets

The data in this paper comes from Weibo (a popular Chinese social networking site). The training corpus for word embedding learning includes 10 million unlabeled blogs of COAE 2014 (The Sixth Chinese Opinion Analysis Evaluation) and 40 thousand blogs labeled using the emotion category in the task of NLPCC2018. The text of Microblog is labeled "none" if it does not convey any emotion. If the text conveys emotion, it is labeled with emotion categories from happiness, trust, anger, sadness, fear, disgust, or surprise. The number of sentences for each emotion category in training data and test data are described in Table 1. We can see from the tables that the distribution of different emotion classes is not balanced.

Table 1. Examples of the emotional vocabulary ontology format.

Emotion Type	Training Data		Test Data	
	Number	Percentage	Number	Percentage
HAPPINESS	5200	13%	3000	15%
TRUST	5200	13%	2200	11%
ANGER	3600	9%	2000	10%
SADNESS	2800	7%	1600	8%
FEAR	2400	6%	1000	5%
DISGUST	4800	12%	2000	10%
SURPRISE	4000	10%	2200	11%
none	12,000	30%	6000	30%

This paper selected DUTIR, which is a Chinese ontology resource collated and labeled by the Dalian University of Technology Information Retrieval Laboratory, as emotional lexicons. There are seven kinds of emotion in the lexical ontology, happiness, trust, anger, sadness, fear, disgust and surprise, a total of 27,466 emotional words, and five emotional intensities, 1, 3, 5, 7, and 9, where 9 indicates the maximum strength and 1 is the least strength. This resource describes a Chinese word or phrase from different perspectives, including lexical categories, emotional categories, emotional intensity, and polarity of words.

In the Chinese Emotion Word Ontology, the general format is shown in Table 2:

Table 2. Examples of the emotional vocabulary ontology format. Part of speech (POS).

Word	POS	Sense	Sense Number	Emotional Subcategory	Strength	Polarity
Happy	adj	2	1	happy	5	1
sad	adj	1	1	sad	5	0

The emotional categories are shown in Table 3.

Table 3. Emotional category.

Emotional Categories	Subcategories	Example Words
HAPPINESS	happy	joy, hilarity
	relieved	relieved, innocent
TRUST	respect	polite, reverent
	praise	handsome, excellent
	believe	trust, reliability
	favorite	admiration, love
	wish	longing for, blessing
ANGER	anger	annoyed, furious
SADNESS	sad	sorrow, heart
	disappointed	regret, despair
	guilty	remorseful, unwilling,
	miss	lovesickness, hangs
FEAR	nervous	flustered, overwhelmed
	dread	timid, scared
	shame	red-faced, self-confident
DISGUST	bored	irritated, upset
	abomination	shameful, hateful
	blame	vanity, chaotic
	jealous	envy
	doubt	suspicious, multi-minded
SURPRISE	surprise	strange, miraculous

4.2. Experimental Results

In this section, we explain the use of the support vector machine (SVM), logistic regression model, decision tree model and gradient boost model classifier to evaluate the effectiveness of the emotional word vector.

The main content is to evaluate the possible advantages of the hybrid word vectors method relative to other word representation methods. The proposed method tries different mixing methods on the emotional word vector and the semantic word vector to compare them according to their classification accuracy on all datasets.

In the emotional classification task, the current widely accepted evaluation indicators are the precision and recall rate. The F-score, a comprehensive metric, was also selected to measure the accuracy of the assessment analysis. The concrete calculation process can be expressed as follows:

$$Precison = \frac{\#system\_correc\,(emotion = i)}{\#system\_proposed\,(emotion = i)} \tag{9}$$

$$Recall = \frac{\#system\_correct\,(emotion = i)}{\#gold\,(emotion = i)} \tag{10}$$

where $\#system\_correc(emotion = i)$ represents the number of texts correctly categorized into a class, and $\#system\_proposed(emotion = i)$ represents the total number of texts classified into a class. The calculation formula for the F-score is described as follows:

$$F\text{-}measure = \frac{2 \times Precison \times Recall}{Precison + Recall}. \tag{11}$$

Experiments use hybrid vectors as input into different classifiers. Here, the SVM, logistic regression classification model, decision tree model and gradient boost model are selected, and the experimental results can be described as follows.

From Table 4, it can be seen that the classification accuracy and reliability of positive emotions such as happiness and trust are higher than the negative emotions such as fear and disgust. The main reason is that in the training corpus, the text scale of positive emotion is larger than the text scale of negative emotion, for example, the number of texts belonging to happiness and trust accounted for a total of 26% of the training data, while the number of texts belonging to sadness and fear accounted for a total of 11%, which indicates that the effect of classification is related to the distribution of training corpus to a certain extent.

Chinese Weibo text content is short with heavy colloquialism and many novel words on the internet, which limits the process of constructing emotional word vectors based on the emotion dictionary. Despite this, the experiment still enhanced the precision, which further proves the validity of the experimental method. The results of this experiment are related to the quality of emotional dictionaries to a certain extent, and emotional words with strong emotions should be chosen in the course of the experiment.

In order to highlight the effectiveness of the emotional word vector, this study carried out comparative experiments on different word representation methods. Specifically, the performance of the method based on the Skip-Gram and that based on the hybrid word vectors proposed in this paper were compared.

Table 4. Performance evaluation sentiment-aware word embedding. Hybrid method 1 is as shown in Equation (6), hybrid method 2 is as shown in Equation (8), Prec is the abbreviation of Precision which is as shown in Equation (9), Rec is the abbreviation of Recall which is as shown in Equation (10), F1 is the abbreviation of F-measure which is as shown in Equation (11).

Model	Method	Evaluation Indicators	HAPPINESS	TRUST	ANGER	SADNESS	FEAR	DISGUST	SURPRISE
Support vector machine (SVM)	Hybrid method 1	Prec	0.7900	0.7976	0.7901	0.6364	0.6579	0.7273	0.6490
		Rec	0.8315	0.7974	0.7807	0.549	0.4902	0.7141	0.9402
		F1	0.8102	0.7973	0.7793	0.5895	0.5161	0.7093	0.7679
	Hybrid method 2	Prec	0.7733	0.7968	0.7873	0.5714	0.5488	0.6933	0.6393
		Rec	0.7858	0.7965	0.7786	0.4706	0.4794	0.6767	0.9316
		F1	0.7795	0.7964	0.7674	0.5161	0.5118	0.6685	0.7583
Logistic Regression	Hybrid method 1	Prec	0.7600	0.8030	0.7742	0.6314	0.6571	0.7146	0.6377
		Rec	0.7600	0.8029	0.7500	0.5092	0.451	0.7079	0.9301
		F1	0.7600	0.8028	0.7605	0.5652	0.5349	0.7050	0.7500
	Hybrid method 2	Prec	0.7451	0.8030	0.7742	0.5713	0.5908	0.7190	0.6492
		Rec	0.7600	0.8029	0.7500	0.4700	0.5908	0.7192	0.9231
		F1	0.7524	0.8028	0.7605	0.5162	0.5475	0.7104	0.7579
decision_tree	Hybrid method 1	Prec	0.6935	0.6966	0.6157	0.6977	0.5116	0.6543	0.6479
		Rec	0.6900	0.6736	0.6709	0.5128	0.4314	0.6500	0.6900
		F1	0.6917	0.7052	0.6421	0.5911	0.4681	0.6521	0.6682
	Hybrid method 2	Prec	0.6624	0.6522	0.6356	0.6989	0.5080	0.6552	0.6531
		Rec	0.6472	0.6571	0.6410	0.5256	0.538	0.6424	0.6531
		F1	0.6547	0.6546	0.6383	0.6000	0.4512	0.6487	0.6531
gradient_boost	Hybrid method 1	Prec	0.8164	0.7975	0.7900	0.6392	0.6453	0.6344	0.6486
		Rec	0.8034	0.7900	0.8125	0.8632	0.6300	0.8675	0.8205
		F1	0.8087	0.7937	0.8009	0.7345	0.6376	0.7329	0.7245
	Hybrid method 2	Prec	0.7906	0.7900	0.7800	0.6486	0.6420	0.6255	0.6012
		Rec	0.7867	0.7900	0.7786	0.8205	0.6372	0.6496	0.8889
		F1	0.7886	0.7900	0.7792	0.7245	0.6395	0.6373	0.7172

As is shown in the Table 5, the experimental results of hybrid word vectors as classifier input are better than the experimental results of the initial word embedding as classifier input. Among them, hybrid method 1 is slightly better than hybrid method 2. The main reason is that mixing the methods increases the difference between the word vectors of different emotions more significantly. Mixing method 2 also brings about the improvement of precision, but the increase is not significant.

Table 5. Accuracy comparison of emotion classification experiments based on various text representations.

Model	Word Vector	Prec	Rec	F1
svm	Skip-gram	0.6685	0.6811	0.6542
	Hybrid word vector1	0.7211	0.7290	0.7099
	Hybrid word vector2	0.6871	0.7027	0.6865
LR	Skip-gram	0.6612	0.6507	0.6674
	Hybrid word vector1	0.7124	0.7015	0.6969
	Hybrid word vector2	0.6932	0.7166	0.6925
decision_tree	Skip-gram	0.6366	0.6100	0.6230
	Hybrid word vector1	0.6453	0.6170	0.6312
	Hybrid word vector2	0.6379	0.6149	0.6143
gradient_boost	Skip-gram	0.6812	0.7511	0.7145
	Hybrid word vector1	0.7102	0.7981	0.7475
	Hybrid word vector2	0.6968	0.7645	0.7251

4.3. Relation to Other Method

Multi-entity sentiment analysis using entity-level feature extraction and word embeddings approach [39] enhance the word embeddings approach with the deployment of a sentiment lexicon-based technique. The paper proposes associating a given entity with the adjectives, adverbs, and verbs describing it and extracting the associated sentiment to try and infer if the text is positive or negative in relation to the entity or entities. We discuss the major differences between Sweeney's model and our method: (i) Lexicon is used in different ways, Sweeney's model uses the lexicon against the parsed text to identify the polarity of the descriptor words (ii) Sweeney's model uses a Twitter-specific parser to identify the descriptor words that relate to a specific entity for text that contains multiple entities. The descriptor words of the multi-entity tweets are scored using SentiWordNet sentiment lexicon. The overall scoring per entity is printed out as output for the multi-entity tweets. The remaining tweets are classified using a random forest classifier. At the end of the article, the author points out the research aims to highlight how a hybrid word embeddings and lexicon-based approach can be used to tackle the problem of sentiment analysis on multiple entities. But we can see that it isolates dictionary information from word embedding techniques and does not use so-called hybrid word embedding techniques. And our approach is the perfect way to achieve this.

We have reproduced the experiment of the paper and compared it with our method:

As is shown in the Table 6, sentiment-aware word embedding is superior to Sweeney's model in the use of external information, and performs better on emotion classification than Sweeney's model. To sum up, the hybrid word vector that combines semantic information with external emotional information not only provides more word feature information but also involves the emotion labeling of the word in the model prediction. The experiments showed that this method is effective, and it is superior to the original model in terms of accuracy, recall rate, and F-score. It can be concluded that in the emotional analysis task, the quality of the word vector can be improved by incorporating external emotional information.

Table 6. Accuracy comparison of Sweeney's model and sentiment-aware word embedding.

Method	Prec	Rec	F1
SentiWordNet + POS	0.66	0.66	0.66
Word2Vec + SentiWordNet + POS	0.6913	0.6754	0.695
Hybrid word vector1	0.7211	0.729	0.7099
Hybrid word vector2	0.6871	0.7027	0.6865

5. Conclusions

Sentiment-aware word embedding was proposed for emotion classification tasks. The method uses mixed emotional word vectors for emotional analysis, through context-sensitive word embedding provided by Word2Vec combinations of the emotional information provided by the dictionary, which is used as the input of the classifier model. The experiment proved that the use of hybrid word vectors is effective for supervised emotion classification, as it greatly improves the accuracy of emotion classification tasks.

In the future, our work will be aimed at doing some experiments in the following directions to demonstrate the flexibility and effectiveness of the method and to further improve its performance: (a) The method will be applied to the other language corpuses to prove its versatility, and (b) novel ways of combining different word vector components will be explored to increase the differentiation of features between two words.

Author Contributions: F.L. conceived the idea, designed and performed the experiments, and analyzed the results, S.C. drafted the initial manuscript, and S.C., J.S., and X.M. revised the final manuscript. R.S. provided experimental environment and academic guidance.

Funding: This work was supported by the National Natural Science Foundation of China (Grant Nos. 61401519, 61872390), the Natural Science Foundation of Hunan Province (Grant Nos. 2016JJ4119, 2017JJ3415), and the Postdoctoral Science Foundation of China (Grant No. 2016M592450).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nikhil, R.; Tikoo, N.; Kurle, S.; Pisupati, H.S.; Prasad, G. A survey on text mining and sentiment analysis for unstructured web data. *J. Emerg. Technol. Innov. Res.* **2015**, *2*, 1292–1296.
2. Huang, E.H.; Socher, R.; Manning, C.D.; Ng, A.Y. Improving word representations via global context and multiple word prototypes. In Proceedings of the Meeting of the Association for Computational Linguistics: Long Papers, Jeju Island, Korea, 8–14 July 2012; pp. 873–882.
3. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv* **2013**, arXiv:1301.3781.
4. Collobert, R.; Weston, J.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural Language Processing (Almost) from Scratch. *J. Mach. Learn. Res.* **2011**, *12*, 2493–2537.
5. Pennington, J.; Socher, R.; Manning, C. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
6. Devlin, J.; Zbib, R.; Huang, Z.; Lamar, T.; Schwartz, R.; Makhoul, J. Fast and Robust Neural Network Joint Models for Statistical Machine Translation. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 23–25 June 2014; pp. 1370–1380.
7. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 3111–3119.
8. Lin, C.-C.; Ammar, W.; Dyer, C.; Levin, L. Unsupervised pos induction with word embeddings. *arXiv* **2015**, arXiv:1503.06760.
9. Turian, J.; Ratinov, L.; Bengio, Y. Word representations: A simple and general method for semi-supervised learning. In Proceedings of the Meeting of the Association for Computational Linguistics, ACL 2010, Uppsala, Sweden, 11–16 July 2010; pp. 384–394.
10. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of Tricks for Efficient Text Classification. *arXiv* **2016**, arXiv:1607.01759.
11. Mesnil, G.; Dauphin, Y.; Yao, K.; Bengio, Y.; Deng, L.; Hakkanitur, D.; He, X.; Heck, L.; Tur, G.; Yu, D. Using Recurrent Neural Networks for Slot Filling in Spoken Language Understanding. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2015**, *23*, 530–539. [CrossRef]
12. Harris, Z.S. *Distributional Structure*; Springer: Dordrecht, The Netherlands, 1981; pp. 146–162.
13. Charles, W.G. Contextual correlates of meaning. *Appl. Psycholinguist.* **2000**, *21*, 505–524. [CrossRef]

14. Rubenstein, H.; Goodenough, J.B. Contextual correlates of synonymy. *Commun. ACM* **1965**, *8*, 627–633. [CrossRef]
15. Zhu, Y.; Min, J.; Zhou, Y. Semantic orientation computing based on HowNet. *J. Chin. Inf. Process.* **2006**, *20*, 14–20.
16. Pan, M.H.; Niu, Y. Emotion Recognition of Micro-blogs Based on a Hybrid Lexicon. *Comput. Technol. Dev.* **2014**, *9*, 6.
17. Cambria, E.; Schuller, B.; Xia, Y.; Havasi, C. New Avenues in Opinion Mining and Sentiment Analysis. *IEEE Intell. Syst.* **2013**, *28*, 15–21. [CrossRef]
18. Turney, P.D. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th annual meeting on association for computational linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 417–424.
19. Lin, C.; He, Y. Joint sentiment/topic model for sentiment analysis. In Proceedings of the 18th ACM Conference on Information and Knowledge Management, Hong Kong, China, 2–6 November 2009; pp. 375–384.
20. Wang, Y.; Youn, H. Feature Weighting Based on Inter-Category and Intra-Category Strength for Twitter Sentiment Analysis. *Appl. Sci.* **2019**, *9*, 92. [CrossRef]
21. Mikolov, T.; Yih, W.-t.; Zweig, G. Linguistic regularities in continuous space word representations. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Atlanta, GA, USA, 9–14 June 2013; pp. 746–751.
22. Liu, Q.; Jiang, H.; Wei, S.; Ling, Z.-H.; Hu, Y. Learning semantic word embeddings based on ordinal knowledge constraints. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 26–31 July 2015; pp. 1501–1511.
23. Faruqui, M.; Dodge, J.; Jauhar, S.K.; Dyer, C.; Hovy, E.; Smith, N.A. Retrofitting Word Vectors to Semantic Lexicons. *arXiv* **2014**, arXiv:1411.4166.
24. Liu, Y.; Liu, Z.; Chua, T.S.; Sun, M. Topical word embeddings. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; pp. 2418–2424.
25. Zhou, C.; Sun, C.; Liu, Z.; Lau, F.C.M. Category Enhanced Word Embedding. *arXiv* **2015**, arXiv:1511.08629.
26. Yu, M.; Dredze, M. Improving Lexical Embeddings with Semantic Knowledge. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 23–25 June 2014; pp. 545–550.
27. Xu, C.; Bai, Y.; Bian, J.; Gao, B.; Wang, G.; Liu, X.; Liu, T.Y. RC-NET: A General Framework for Incorporating Knowledge into Word Representations. In Proceedings of the ACM International Conference on Conference on Information and Knowledge Management, Shanghai, China, 3–7 November 2014; pp. 1219–1228.
28. Levy, O.; Goldberg, Y. Dependency-Based Word Embeddings. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 23–25 June 2014; pp. 302–308.
29. Lu, A.; Wang, W.; Bansal, M.; Gimpel, K.; Livescu, K. Deep Multilingual Correlation for Improved Word Embeddings. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 31 May–5 June 2015; pp. 250–256.
30. Hermann, K.M.; Blunsom, P. Multilingual Models for Compositional Distributed Semantics. *arXiv* **2014**, arXiv:1404.4641.
31. Zhang, J.; Liu, S.; Li, M.; Zhou, M.; Zong, C. Bilingually-constrained phrase embeddings for machine translation. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 23–25 June 2014; pp. 111–121.
32. Ren, F.; Deng, J. Background Knowledge Based Multi-Stream Neural Network for Text Classification. *Appl. Sci.* **2018**, *8*, 2472. [CrossRef]
33. Wang, Y.; Wang, S.; Tang, J.; Liu, H.; Li, B. Unsupervised sentiment analysis for social media images. In Proceedings of the IEEE International Conference on Data Mining Workshop, Washington, DC, USA, 14–17 November 2015; pp. 1584–1591.
34. Hogenboom, A.; Bal, D.; Frasincar, F.; Bal, M.; Jong, F.D.; Kaymak, U. Exploiting emoticons in sentiment analysis. In Proceedings of the 28th Annual ACM Symposium on Applied Computing, Coimbra, Portugal, 18–22 March 2013; pp. 703–710.

35. Hu, X.; Tang, J.; Gao, H.; Liu, H. Unsupervised sentiment analysis with emotional signals. In Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 13–17 May 2013; pp. 607–618.
36. Tang, D.; Wei, F.; Qin, B.; Yang, N.; Liu, T.; Zhou, M. Sentiment Embeddings with Applications to Sentiment Analysis. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 496–509. [CrossRef]
37. Carrillo-De-Albornoz, J.; Plaza, L. An emotion-based model of negation, intensifiers, and modality for polarity and intensity classification. *J. Assoc. Inf. Sci. Technol.* **2013**, *64*, 1618–1633. [CrossRef]
38. Chen, J. The Construction and Application of Chinese Emotion Word Ontology. Master's Thesis, Dailian University of Technology, Dalian, China, 2008.
39. Sweeney, C.; Padmanabhan, D. Multi-entity sentiment analysis using entity-level feature extraction and word embeddings approach. In Proceedings of the Recent Advances in Natural Language Processing, Varna, Bulgaria, 4–6 September 2017.