

Received June 25, 2019, accepted July 5, 2019, date of publication July 23, 2019, date of current version August 2, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2929204

Tree-Structured Neural Networks With Topic Attention for Social Emotion Classification

CHANG WANG¹, BANG WANG¹, AND MINGHUA XU²

¹School of Electronic Information and Communications, Huazhong University of Science and Technology (HUST), Wuhan 430074, China

²School of Journalism and Information Communication, Huazhong University of Science and Technology (HUST), Wuhan 430074, China

Corresponding author: Minghua Xu (xuminghua@hust.edu.cn)

This work was supported in part by the National Social Science Foundation of China under Grant 19BXW051.

ABSTRACT Social emotion classification studies the emotion distribution evoked by an article among numerous readers. Although recently neural network-based methods can improve the classification performance compared with the previous word-emotion and topic-emotion approaches, they have not fully utilized some important sentence language features and document topic features. In this paper, we propose a new neural network architecture exploiting both the syntactic information of a sentence and topic distribution of a document. The proposed architecture first constructs a tree-structured long short-term memory (Tree-LSTM) network based on the sentence syntactic dependency tree to obtain a sentence vector representation. For a multi-sentence document, we then use a Chain-LSTM network to obtain the document representation from its sentences' hidden states. Furthermore, we design a topic-based attention mechanism with two attention levels. The word-level attention is used for weighting words of a single-sentence document and the sentence-level attention for weighting sentences of a multi-sentence document. The experiments on three public datasets show that the proposed scheme outperforms the state-of-the-art ones in terms of higher average Pearson correlation coefficient and MicroF1 performance.

INDEX TERMS Long short-term memory, social emotion classification, topic attention mechanism, topic model, tree-structured neural network.

I. INTRODUCTION

Social emotion classification is to study the evoked emotion distribution among a great number of readers who have read a same article [1]. In some news websites, readers can mark their emotion as one of the following ones, 'touch', 'surprise', 'amusement', 'sadness', 'curiosity', and 'anger', after reading a piece of news, and the social emotion distribution is displayed as a vote histogram at the end of a news article. Fig. 1 illustrates such a social emotion distribution of one article in a well-known Chinese news website, Sina News.¹ Understanding and predicting social emotion distribution for one article or news are envisioned with lots of applications, such as article narrative classification, online information diffusion, public opinion monitoring and etc. [2]–[4].

As a sub-task of sentiment analysis, the earliest research of social emotion classification appeared in the SemEval-2007 task 14 [5]. The early methods are word-emotion models, which have focused on the features of individual words to

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

¹<https://news.sina.com.cn>

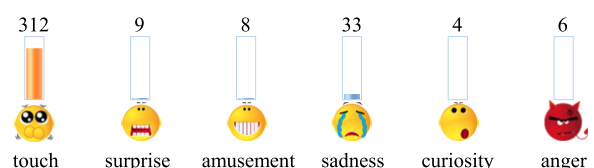


FIGURE 1. Social emotions displayed as a vote histogram in Sina News.

find direct relations between words and emotions [1], [2], [6]. Such word-emotion models cannot distinguish different emotions of a same word in different contexts. Later on, some topic-emotion models have been proposed. They utilize a topic model like *Latent Dirichlet Allocation* (LDA) to discover the topical information of a document and model the relation between words and emotions [1], [7]–[11]. However, both the word-emotion models and topic-emotion models treat individual word separately without considering the semantic information in between words of a sentence.

Recently, with the successive application of neural network technology in some natural language processing

tasks, such as neural language models [12]–[14], text classification [15] and sentiment classification [16], some neural network-based methods have been proposed for social emotion classification [17], [18]. They adopt a *convolutional neural network* (CNN) or a *recurrent neural network* (RNN) to learn the semantic features (like the temporal order) and obtain a vector representation of the document. Although the neural network is a powerful tool to obtain document representation, these work have not fully utilized some important language and document features, like the syntactic information of a sentence and topical distribution of a document.

In this paper, we study the problem of social emotion classification by designing a new neural network architecture for obtaining document representation. We argue that the syntactic dependency relation in between words of a sentence is also an important feature for a sentence, which actually have been verified in many related tasks like semantic relatedness [19]–[23]. Furthermore, we also support the claim in the existing work that the document topical distribution would help to distinguish the emotion of a same word in different contexts. Instead of establishing an emotion-topic model, we propose to include the document topical distribution as a kind of attention mechanism into the proposed neural network. The attention mechanism is to assign different components different importance weights [24], [25], which has been successively applied in many neural networks for machine translation and document classification tasks [26]–[28].

In this paper, we propose a two-layer neural network structure with a LDA-based attention mechanism for social emotion classification. For a document consisting of more than one sentence, the lower layer is implemented with a Tree-LSTM network (*tree-structured long short-term memory network*), which encodes the syntactic feature of each sentence into a sentence vector. The upper layer is a Chain-LSTM (chain-structured LSTM) with sentence-level LDA attention, where the Chain-LSTM is used to obtain the hidden state of each sentence, and our sentence-level LDA attention is computed as the similarity between each sentence topic distribution and the document topic distribution. We then use the attention weights to weight the sum of sentence hidden states for obtaining the document representation. For a document with a single sentence, the proposed structure only contains the lower-layer Tree-LSTM network, yet the word-level LDA attention is computed as the similarity between each word topic distribution and the document topic distribution. After obtaining document representation, we input it into a softmax layer to obtain the final social emotion distribution. Experimenting on three public datasets, the results have validated the superiority of our proposed scheme over the state-of-the-art ones in terms of higher average Pearson correlation coefficient and MicroF1 performance.

The main contributions of our work are as follows:

- We propose a hierarchical neural network structure to obtain document vector representation.

- We propose to use a Tree-LSTM network to encode syntactic dependency for words in a sentence and integrate a LDA-based attention mechanism into the neural network, to simultaneously exploit syntactic and topic information for social emotion classification.
- We experiment and compare the proposed scheme over three public datasets.

The remainder of this paper is organized as follows: Section II briefly reviews the related work. The proposed scheme is presented in Section III and experimented in Section IV. The paper is concluded in Section V.

II. RELATED WORK

A. SOCIAL EMOTION CLASSIFICATION

The study of social emotion classification could be dated back to the SemEval-2007 task 14 [5]. Since then, three main solution approaches can be identified, namely, word-emotion, topic-emotion and neural network. The word-emotion methods are based on the handcrafted features of individual words for classification [1], [2], [6]. For example, the SWAT system [6] first establishes a word-emotion mapping dictionary, which is used to score each word in an unlabeled news headline and derive the overall emotion score. The emotion-term model [1] treats individual words as independently generated from social emotion labels, and then finds the relation in between words and social emotions with a Bayesian approach. However, such word-emotion models ignore the fact that a same word could convey different emotions in different contexts.

The topic-emotion approaches try to distinguish word emotions in different contexts by establishing the relations in between the document topic distribution and social emotions [1], [7]–[11]. For example, the emotion-topic model [1] has introduced an emotion layer into the LDA topic model to jointly model topics and emotions. The affective topic model [8] has also designed an intermediate layer into the LDA model to associate each topic with words' emotions.

Recently, the neural network approaches have gained lots of focuses for its capable of learning and extracting hidden semantic information of a document [17], [18], [29]. For example, Zhao *et al.* [17] have proposed a parallel network of Bidirectional LSTM (BiLSTM) and a convolution neural network (CNN) to obtain a document representation as the concatenation of the output vectors of the two networks. Li *et al.* [18] have modeled one document as a single sequence and designed a CNN network which contains a word-level and a phrase-level convolution layer to model word-phrase and phrase-sentence relation respectively. A phrase represents a nugget of words in the document. Li *et al.* [29] have proposed a hybrid neural network that leverages semantic domain knowledge from unsupervised teaching models like Bi-Term Topic Model (BTM), Replicated Softmax Machine (RSM) or Word2vec. However, these neural networks have ignored the syntactic information of a sentence.

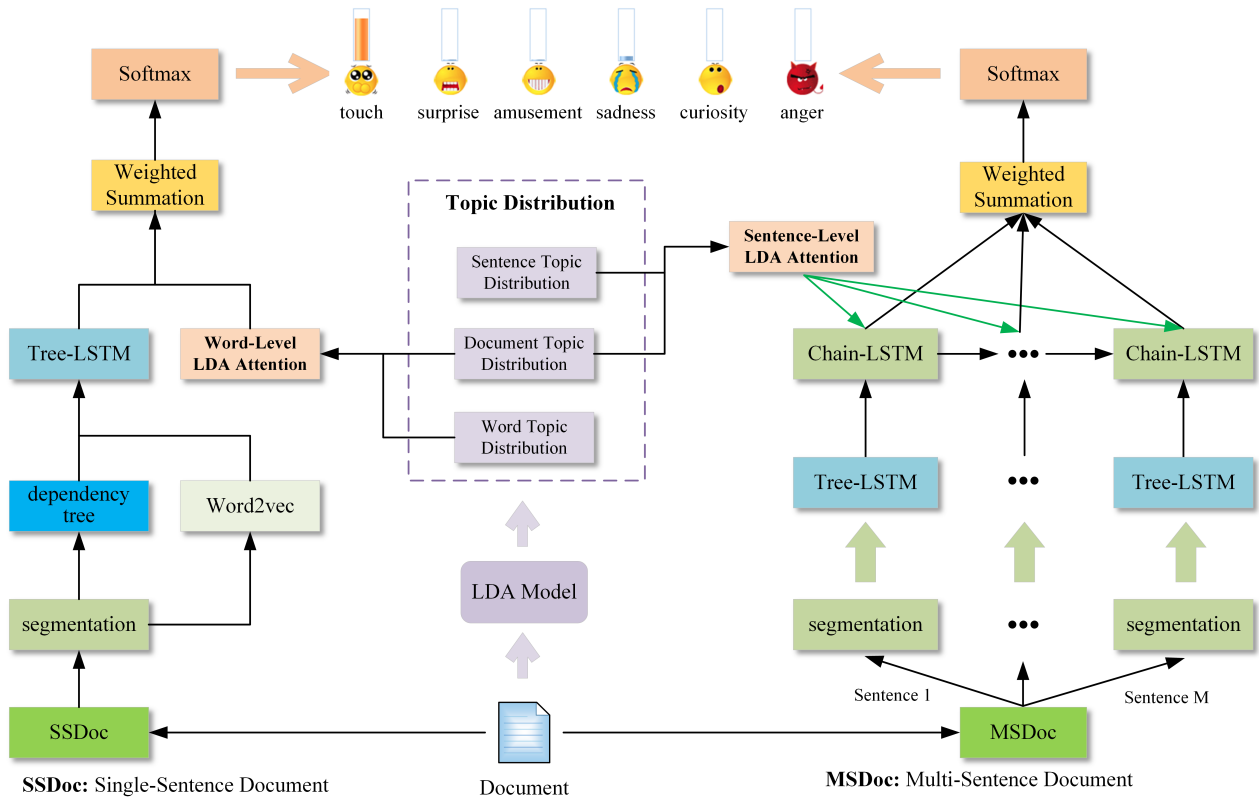


FIGURE 2. The framework of the proposed solution: An input document is first classified as a single-sentence document (SSDoc) or a multi-sentence document (MSDoc). In the sentence level, we perform the word segmentation for Chinese sentence to obtain the composing words and use Word2Vec to obtain each word embedding. We next perform a syntactic analysis to obtain the dependency tree for each sentence and construct a Tree-LSTM network accordingly. For SSDoc, we compute the word-level LDA attention from the LDA model to obtain the weighted summation of words’ hidden vectors as the document representation. For MSDoc, the lower-layer is also a Tree-LSTM network yet without attention. We use a Chain-LSTM together with sentence-level LDA attention to obtain a multi-sentence documentation representation. Based on the document representation, we use a linear layer together with the softmax to output the emotion distribution of the document.

B. ATTENTION MECHANISM

Attention mechanism, which can guide the network to unequally treat each component of the input according to its importance, has been shown the capability to improve the performance of neural networks in many natural language processing tasks [26]–[28]. For example, Yang *et al.* [27] have proposed a hierarchical attention network (HAN) consisting of a layer of word encoder and a layer of sentence encoder, each of which has been implemented by a BiGRU (Bidirectional Gated Recurrent Units) with an attention mechanism. The network can extract the important words or sentences automatically for document classification. Kokkinos and Potamianos [28] have proposed a tree-structured BiGRU network with attention for sentence-level sentiment classification, where the attention mechanism is used to learn the importance of words in a sentence. Different from these work where the attention mechanism is MLP-based (multi-layer perceptron), we explore a novel LDA-based attention mechanism in this paper.

III. THE PROPOSED SOLUTION

Our neural network models leverage sentence syntactic information via dependency analysis and integrate topical

attention via LDA analysis. In social emotion classification, a target document can consist of a single sentence or multiple sentences. We design different neural networks for the two kinds of documents: For a *single-sentence document* (SSDoc), we obtain the SSDoc representation using a Tree-LSTM structure with the Word-level LDA-Attention mechanism (Section 4.1). For a *multi-sentence document* (MSDoc), we first obtain each sentence vector by a Tree-LSTM structure, and then feed sentence vectors into a Chain-LSTM structure with the Sentence-level LDA-Attention mechanism to generate the MSDoc representation. Commonly, after document modeling, a softmax output layer is used for social emotion classification.

The network framework is shown in Fig. 2, where the left part is for SSDoc and the right part is for MSDoc social emotion classification. At first, an input document is decided as a SSDoc or a MSDoc. As a Chinese sentence does not contain natural word delimitations, we then apply a word segmentation technique to divide one sentence into multiple words.² For each word, we use a pre-trained Word2vec model [12] to obtain a word embedding, which is a low dimensional dense vector of real numbers. In this paper, we use w , s and d

²<https://github.com/fxsjy/jieba>

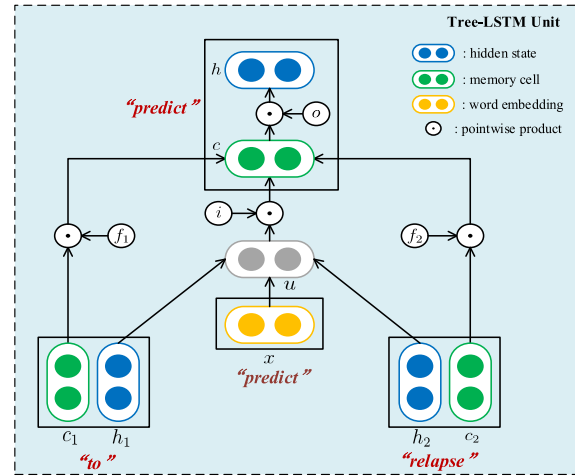
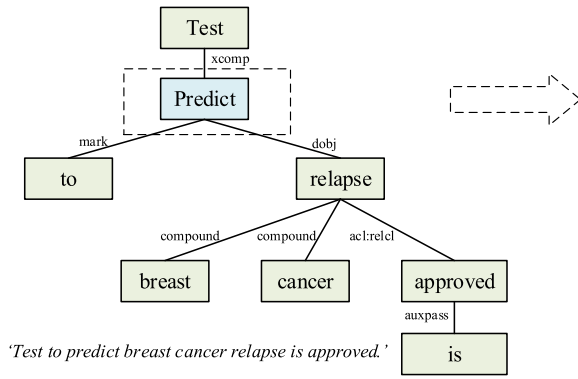


FIGURE 3. An example of dependency tree produced by the Stanford Parser. The nodes represent words. The edges represent syntactic relations. The raw sentence is “Test to predict breast cancer relapse is approved”.

to denote a word embedding, sentence vector and document vector, respectively.

A. VECTOR REPRESENTATION FOR SINGLE-SENTENCE DOCUMENT

1) TREE-LSTM

We propose to use a Tree-LSTM network to obtain a *hidden vector* for each word of a sentence. To build a Tree-LSTM, we first apply a dependency analysis tool, Stanford Parser [30] (for English) or LTP [31] (for Chinese), to obtain the dependency tree for one sentence, where each node represents a word and the syntactic dependency relations are captured by the tree edges.

The left part of Fig. 3 presents the dependency tree for a sentence ‘Test to predict breast cancer relapse is approved’. We can see that each word is modeled as a node in a five-layer tree structure, where a parent node is connected with one or more child nodes and each parent-child relation is with some syntactic dependency. By modeling each node as a Tree-LSTM unit, we construct a Tree-LSTM network to obtain each word a hidden state vector. Specifically, a Tree-LSTM unit accepts the hidden state(s) of its child node(s) and the word embedding of the current node to compose the hidden state of the current node. The hidden states of the child nodes for leaf nodes are initialized to zero vectors. The process follows the dependency structure, starting from the bottom leaf nodes till the root node.

The right part of Fig. 3 illustrates the internal structure of a Tree-LSTM unit. For one Tree-LSTM unit, let $C(w_n)$ denote the set of its child word(s). The transition functions of the Tree-LSTM unit are calculated as follows:

$$\begin{aligned} \mathbf{f}_j &= \sigma(\mathbf{W}^{(f)} \mathbf{x}_n + \mathbf{U}^{(f)} \mathbf{h}_j + \mathbf{b}^{(f)}), \quad j \in C(w_n) \\ \mathbf{i}_n &= \sigma(\mathbf{W}^{(i)} \mathbf{x}_n + \mathbf{U}^{(i)} \tilde{\mathbf{h}}_n + \mathbf{b}^{(i)}) \\ \mathbf{o}_n &= \sigma(\mathbf{W}^{(o)} \mathbf{x}_n + \mathbf{U}^{(o)} \tilde{\mathbf{h}}_n + \mathbf{b}^{(o)}) \end{aligned}$$

$$\begin{aligned} \mathbf{u}_n &= \tanh(\mathbf{W}^{(u)} \mathbf{x}_n + \mathbf{U}^{(u)} \tilde{\mathbf{h}}_n + \mathbf{b}^{(u)}) \\ \mathbf{c}_n &= \mathbf{i}_n \odot \mathbf{u}_n + \sum_{j \in C(w_n)} \mathbf{f}_j \odot \mathbf{c}_j \\ \mathbf{h}_n &= \mathbf{o}_n \odot \tanh(\mathbf{c}_n) \end{aligned}$$

where $\tilde{\mathbf{h}}_n = \sum_{j \in C(w_n)} \mathbf{h}_j$. The hidden state $\mathbf{h}_n \in \mathbb{R}^{d_m}$, the memory cell $\mathbf{c}_n \in \mathbb{R}^{d_m}$, where d_m is the memory unit dimensionality. We use \odot to denote the pointwise product operation. We can see that the gate functions and the hidden state of a node are both dependent on the state of its child node(s), which can exploit the syntactic information of a sentence.

2) WORD-LEVEL LDA ATTENTION

The attention mechanism is to assign different importance for each individual component when composing components into a single embedding. In the literature, attention mechanism has been applied for text classification and sentiment classification, yet the attention weights are obtained with a multilayer perceptron (MLP) in the neural network [27], [28]. In this paper, we propose to use word-level LDA attention to weight each word hidden state to obtain the SSDoc vector representation. We argue that the topical information is important in the social emotion classification task, as an article usually focuses on some specific topics, yet readers’ emotional reactions might often be related to particular topics described in an article. To this end, we exploit the LDA topic model [32] to obtain word-level attention.

According to the LDA model, a document is considered as a mixture over various latent topics and each latent topic is a probability distribution over all the words. We first train the LDA model with a training corpus. Then for an input document, the LDA model can output the topic probability distribution of the document, denoted by $\mathbf{p}_d = \{\Pr(k|d)\}_{k=1}^K$, and the topic probability distribution of each word in the document, denoted by $\mathbf{p}_{w_n} = \{\Pr(k|w_n)\}_{k=1}^K$, where k represents

the topic index and K the number of topics. We use the cosine distance with a scale of $[0, 1]$ between a word \mathbf{p}_{w_n} and the document \mathbf{p}_d to compute the topical similarity between them, which is then normalized to get the word-level LDA attention weight:

$$\text{sim}(w_n, d) = \frac{\mathbf{p}_{w_n} \cdot \mathbf{p}_d^\top}{|\mathbf{p}_{w_n}| |\mathbf{p}_d|}, \quad (1)$$

$$\alpha_{w_n} = \frac{0.5 \times \text{sim}(w_n, d) + 0.5}{\sum_{n'=1}^N [0.5 \times \text{sim}(w_{n'}, d) + 0.5]}, \quad (2)$$

where N is the number of words in a sentence. Note that stop-words, regarded as the words without topic information, are not considered in the LDA model. So they have no topic distributions and get zero attention weight.

Based on the word hidden state vector \mathbf{h}_n and its topical attention weight α_{w_n} , we compute the SSDoc representation \mathbf{d}_{ss} as the sentence vector \mathbf{s} as follows:

$$\mathbf{d}_{ss} = \sum_{n=1}^N \alpha_{w_n} \mathbf{h}_n.$$

We argue that the vector \mathbf{d}_{ss} not only contains sentence syntactic information as from word hidden vectors, but also includes word topical attention weights.

B. VECTOR REPRESENTATION FOR MULTI-SENTENCE DOCUMENT

We treat a multi-sentence document as a sequence of sentences and obtain a MSDoc representation in two stages. In the first stage, we model each sentence by a Tree-LSTM network in a similar way as that for single-sentence document. The difference is that instead of using the word-level LDA attention in composing a SSDoc vector, we use the hidden state of the root node as the sentence vector directly for simplifying the network structure and avoiding the reuse of topical information in the upstream document vector. Let $\mathbf{s}_m, m = 1, \dots, M$, denote the m th sentence vector in a multi-sentence document.

In the second stage, a Chain-LSTM network with sentence-level LDA attention is employed to transform sentence vectors into a MSDoc vector. Using the Chain-LSTM can help to capture potential inter-sentence relations (such as ‘‘cause’’ and ‘‘contrast’’) into the document representation [16]. The transition functions of a Chain-LSTM unit are as follows:

$$\begin{aligned} \mathbf{f}_m &= \sigma(\mathbf{W}^{(f)} \mathbf{s}_m + \mathbf{U}^{(f)} \mathbf{h}_{m-1} + \mathbf{b}^{(f)}) \\ \mathbf{i}_m &= \sigma(\mathbf{W}^{(i)} \mathbf{s}_m + \mathbf{U}^{(i)} \mathbf{h}_{m-1} + \mathbf{b}^{(i)}) \\ \mathbf{o}_m &= \sigma(\mathbf{W}^{(o)} \mathbf{s}_m + \mathbf{U}^{(o)} \mathbf{h}_{m-1} + \mathbf{b}^{(o)}) \\ \mathbf{u}_m &= \tanh(\mathbf{W}^{(u)} \mathbf{s}_m + \mathbf{U}^{(u)} \mathbf{h}_{m-1} + \mathbf{b}^{(u)}) \\ \mathbf{c}_m &= \mathbf{i}_m \odot \mathbf{u}_m + \mathbf{f}_m \odot \mathbf{c}_{m-1} \\ \mathbf{h}_m &= \mathbf{o}_m \odot \tanh(\mathbf{c}_m) \end{aligned}$$

where the forget gate \mathbf{f}_m , the input gate \mathbf{i}_m , the output gate \mathbf{o}_m control the flow of memory from the previous moment to the

next moment. And $\mathbf{h}_m \in \mathbb{R}^{d_m}$ and $\mathbf{c}_m \in \mathbb{R}^{d_m}$ denote the hidden state and the memory cell in the m th sentence, respectively.

1) SENTENCE-LEVEL LDA ATTENTION

For a multi-sentence document, we argue that not every sentence is equally important for the composition of MSDoc vector representation. Furthermore, we argue that the more similar between a sentence and the whole document in terms of their topic distributions, the more important of the sentence. As such, we propose the following sentence-level LDA attention to weight each sentence and aggregate the weighted sentence vectors to form a MSDoc vector.

Notice that the LDA model does not directly output the topic probability distribution for each individual sentence $\mathbf{p}_{s_m} = \{\Pr(k|s_m)\}_{k=1}^K$, but can output the probability of one word appearing in each topic $\Pr(w|k)$. So we first compute the topic probability of one sentence $\Pr(k|s_m)$ from its composing words' topic probability $\Pr(w|k)$:

$$\Pr(k|s_m) = \frac{q(k|s_m)}{\sum_{k=1}^K q(k|s_m)}, \quad (3)$$

$$q(k|s_m) = \frac{\sum_{w \in s_m} \Pr(w|k) \Pr(k|d)}{N_s} \quad (4)$$

where N_s is the number of words in the sentence. We next compute the topical similarity between a sentence s_m and the document d by applying the *information radius* (IR) based on the Kullback-Liebler (KL) divergence as follows:

$$\text{sim}(s_m, d) = 10^{-IR(\mathbf{p}_{s_m}, \mathbf{p}_d)}, \quad (5)$$

where

$$IR(\mathbf{p}_{s_m}, \mathbf{p}_d) = KL(\mathbf{p}_{s_m} || \frac{\mathbf{p}_{s_m} + \mathbf{p}_d}{2}) + KL(\mathbf{p}_d || \frac{\mathbf{p}_{s_m} + \mathbf{p}_d}{2}).$$

A large similarity $\text{sim}(s_m, d)$ generally indicates that the sentence is important for conveying document topic information. We then compute the sentence-level LDA attention as follows:

$$\alpha_{s_m} = \frac{e^{\text{sim}(s_m, d)}}{\sum_{m'=1}^M e^{\text{sim}(s_{m'}, d)}}, \quad (6)$$

where M is the number of sentences in the document.

Based on the sentence hidden state vectors \mathbf{h}_m and its topical attention weight $\alpha_{s_m}, m = 1, \dots, M$, we then compute the MSDoc vector \mathbf{d}_{ms} as follows:

$$\mathbf{d}_{ms} = \sum_{m=1}^M \alpha_{s_m} \mathbf{h}_m$$

We argue that the MSDoc representation \mathbf{d}_{ms} not only contains the syntactic information of individual sentences from their composing words by the lower-layer Tree-LSTM network, but also includes the topic information of the whole document from its composing sentences yet weighted by the sentence-level LDA attention.

C. SOCIAL EMOTION CLASSIFICATION

1) OUTPUT

After obtaining the document representation \mathbf{d}_{ss} or \mathbf{d}_{ms} , we apply a linear layer to transform it into a label vector $\mathbf{z} = \{z_l\}_{l=1}^E$, where E is the number of emotion labels. We then normalize \mathbf{z} by another softmax layer to output the predicted final emotion classification vector $\hat{\mathbf{y}} = \{\hat{y}_l\}_{l=1}^E$ as follows:

$$\hat{y}_l = \frac{e^{z_l}}{\sum_{l=1}^E e^{z_l}}, \quad \mathbf{z} = \mathbf{W}^{(t)} \mathbf{d}, \quad (7)$$

where $\mathbf{W}^{(t)}$ is the weight matrix for linear layer.

2) TRAINING

We note that in our task of social emotion classification, the evaluation objective is an emotion probability distribution, other than a single most likely emotion label. So in the training phase, we use the Kullback-Leibler divergence between the ground truth emotion distribution \mathbf{y} and the predicted emotion distribution $\hat{\mathbf{y}}$ as the loss function:

$$\text{Loss}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{E} \sum_{l=1}^E y_l (\log(y_l) - \log(\hat{y}_l)) \quad (8)$$

where \hat{y}_l and y_l represent the predicted probability and the true probability of the l th emotion label respectively. We train the whole network with the Adam optimizer.

IV. EXPERIMENT RESULTS

A. EXPERIMENT DATASETS

We use three public datasets for our experiments: The dataset **SinaNews**, [33] contains only multi-sentence documents; The dataset **SemEval** [5] contains only single-sentence documents; And the dataset **ISEAR** [34] contains both types of documents. For the **ISEAR** dataset, we regarded all documents as a multi-sentence document to ensure the unification of the network framework (even sometimes a document only containing one sentence). This is to examine the adaptiveness of the proposed scheme.

SinaNews [33]: It is a Chinese dataset, which consists of 5258 pieces of hot news collected from the social channel of the news website (*www.sina.com*) from January to December 2016. For each news, it includes the headline, the news body and the user ratings over 6 emotion labels: 'anger', 'touch', 'sadness', 'amusement', 'curiosity' and 'surprise'. On average, each news article was voted by 770.41 users. To be consistent with the baseline methods [33], we use the 3109 articles published from January to June as the training dataset and the 2149 articles published from July to November as the testing dataset.

SemEval [5]: It was provided by the SemEval-2007 task 14, which contains 1250 English news headlines extracted from news web sites (such as Google news, CNN) and newspapers. Each headline is annotated by emotion scores of [0, 100] over 6 emotions: 'anger', 'disgust', 'fear', 'joy', 'sadness' and 'surprise' (0 = the emotion is not present in the headline; 100 = the maximum). After removing 4 samples

without scores, 1000 headlines were used as the training dataset and 246 headlines were used as the testing dataset, which is the same experiment setting as that in [18].

ISEAR [34]: It is an English dataset with 7666 samples, where each sample contains a paragraph of text labeled by one of the following 7 emotions: 'joy', 'fear', 'anger', 'sadness', 'disgust', 'shame' and 'guilt'. Different from the **SinaNews** and **SemEval**, it was tagged with a single label, not a distribution of emotion intensities. Like [29], we randomly select 60% of the samples as the training dataset and the remaining 40% as the testing dataset.

TABLE 1. Statistics of the three datasets.

	labels	#samples	proportion	#votes/valences
SinaNews	anger	1,925	0.37	1,419,477
	touch	829	0.16	489,027
	sadness	683	0.13	561,504
	amusement	1,504	0.29	926,905
	curiosity	109	0.02	200,697
	surprise	208	0.04	453,218
SemEval	anger	87	0.07	12,042
	disgust	42	0.03	7,634
	fear	194	0.16	20,306
	joy	441	0.35	23,613
	sadness	265	0.21	24,039
	surprise	217	0.17	21,495
ISEAR	joy	1,094	0.14	1,094
	fear	1,095	0.14	1,095
	anger	1,096	0.14	1,096
	sadness	1,096	0.14	1,096
	disgust	1,096	0.14	1,096
	shame	1,096	0.14	1,096
	guilt	1,093	0.14	1,093

Table. 1 presents the details of the three datasets, where #samples denotes the number of samples with the highest vote of each emotion label, #votes/valence represents the total number of votes/valences for each emotion label.

B. PARAMETER SETTING

For training the Word2vec model, we resort to public corpus, since the three datasets are too small. For the Chinese dataset **SinaNews**, we used the Chinese Wikipedia corpus³ to train a Chinese Word2vec model with SkipGram algorithm [12]. For the two English datasets **SemEval** and **ISEAR**, we directly used the 300-dimensional English Word2vec model provided by Google.⁴

For obtaining the dependency tree, we use the Stanford Parser [30] for the two English datasets. For the Chinese dataset, we used the LTP toolkit provided by HIT [31] to perform sentence splitting and dependency tree construction. Since the **SemEval** contains only 1250 samples, we resort to another news headline dataset **ABCnews**, [35] for the LDA model training. The **ABCnews** dataset contains 1,103,665 headlines published from 2003-02-19 to 2017-12-31 from Australian Broadcasting Corporation.⁵ Considering that **SemEval** was established in 2007, only 355,151 samples

³<https://dumps.wikimedia.org/>

⁴<https://code.google.com/archive/p/word2vec/>

⁵<http://www.abc.net.au/>

TABLE 2. Parameters for our methods.

	Parameters	SinaNews	SemEval	ISEAR
Word2vec	d_w	100	300	300
LDA	K	30	30	30
networks	d_m	200	300	300
	learning rate	0.001	0.001	0.001
	batch size	20	20	20

before 2007 (including 2007) in *ABCnews* were used as the LDA training corpus.

Table 2 presents the detailed parameter settings, where d_w denotes the dimensionality of word embedding, K denotes the number of topics when training LDA models, and d_m is the dimensionality of memory cell and hidden state in Tree-LSTMs and Chain-LSTMs.

C. EVALUATION METRICS

We adopt the widely used performance metrics for our experiments: the Micro-averaged F1 score, denoted as *MicroF1*, and the average Pearson correlation coefficient, denoted as *AP*. The former can reflect the accuracy of the predicted top-ranked emotion among all emotion labels; While the latter can measure the difference between the predicted emotion probability distribution and the actual distribution.

We denote the actual top-ranked emotion triggered by an article d as e_{top}^d , and denote the predicted emotion as \hat{e}_{top}^d . If there are emotions with the same probability in the actual emotion distribution, their positions are interchangeable. According to [29], we also compute the *MicroF1* as follows:

$$MicroF1 = \frac{\sum_{d \in \mathcal{D}_{test}} \mathbb{I}_d}{|\mathcal{D}_{test}|}$$

where

$$\mathbb{I}_d = \begin{cases} 1, & \text{if } \hat{e}_{top}^d = e_{top}^d, \\ 0, & \text{otherwise,} \end{cases}$$

and \mathcal{D}_{test} is the testing dataset. A larger *MicroF1* indicates the system performs better in predicting the top-ranked emotion label.

According to [33], we compute *AP* as follows:

$$AP = \frac{\sum_{\mathcal{D}_{test}} r(\hat{\mathbf{y}}, \mathbf{y})}{|\mathcal{D}_{test}|},$$

where

$$r(\hat{\mathbf{y}}, \mathbf{y}) = \frac{cov(\hat{\mathbf{y}}, \mathbf{y})}{\sqrt{var(\hat{\mathbf{y}})var(\mathbf{y})}}.$$

r is the Pearson correlation coefficient between the predicted emotion distribution $\hat{\mathbf{y}}$ and the ground truth distribution \mathbf{y} , and cov denotes the covariance operation. *AP* ranges in $[-1, +1]$. The closer the *AP* to 1, the more effective the prediction.

D. COMPARISON SCHEMES

We give a brief description of the peer schemes for performance comparison: Some of them are from the literature; While some are the schemes designed by ourselves for more comprehensive comparison.

The following peer schemes are from the literature:

1) SWAT

It was once the best method proposed in the SemEval-2007 task 14 [36], which creates word-emotion mapping with a corpus of hand annotated headlines and then scores emotions of each headline in the testing dataset by averaging the emotion score of each headline word.

2) EMOTION TERM MODEL (ET) [1]

It assumes that words are independently generated from social emotion labels. Following the Bayesian method, it models the association between word and social emotion.

3) EMOTION TOPIC MODEL (ETM) [1]

It constructs an emotion-topic model by adding an additional layer for emotion modeling into the LDA model. As a result, affective terms and social emotions are combined to complement social emotion classification.

4) WEIGHTED MULTI-LABEL CLASSIFICATION MODEL (WMCM) [11]

It introduces a concept of ‘‘emotional concentration’’ to compute the weight of documents for each emotion. A topic model is used to estimate the joint probability of the document and each word. Finally the social emotion of a document is inferred by the Bayesian theory.

5) CONTEXTUAL SENTIMENT TOPIC MODEL (CSTM) [9]

It aims at distinguishing explicitly generalized topics that are context-independent from both a background theme and a contextual theme.

6) SOCIAL OPINION MINING MODEL (SOM) [33]

It constructs a social opinion network based on the opinion distance computed by a word mover distance algorithm. The social opinion of a new sample is computed based on the nearest neighbor analysis.

7) 1-HNN-BTM [29]

It designs a hybrid neural network which incorporates the biterm topic model for social emotion classification.

8) WEIGHTED PCNN [18]

It is a hierarchical CNN network with both word- and phrase-level convolution. It also employs an ‘emotional concentration’ indicator to weight documents in the training phase to reduce the impact of noisy instances.

9) CNN,PORIA2015DEEP AND CNN-SVM [38]

The CNN network was proposed in [37] which includes 7 layers: 1 input, 2 convolution, 2 max-pooling, 1 fully connected and 1 softmax output layer. The CNN method directly uses the classification result of the CNN network. The CNN-SVM method treats the output of the fully connected layer in the trained CNN as the feature vector of the input document. Then the feature vectors are used to train a SVM classifier. The results of these two methods are from [33].

We also design the following experimental models to more clearly verify the effectiveness of the syntactic information and the LDA-based attention mechanism.

10) HIERARCHICAL LSTM (HLSTM)

It is a hierarchical structure of two LSTM networks. A lower layer LSTM network is used to obtain each sentence vector from its word embedding. Then sentence vectors are fed into an upper layer LSTM network for composing the document representation.

11) HIERARCHICAL LSTM WITH WORD-LEVEL ATTENTION (HLSTM-WA)

We include a general attention mechanism based on MLP [27] into the lower layer LSTM network of HLSTM when composing sentence vectors.

12) HIERARCHICAL LSTM WITH SENTENCE-LEVEL ATTENTION (HLSTM-SA)

We include a general attention mechanism based on MLP [27] into the upper layer LSTM network of HLSTM when composing document representation.

13) HIERARCHICAL LSTM WITH WORD-LEVEL LDA ATTENTION (HLSTM-WA-LDA)

We add Word-level LDA Attention into the first stage for composing sentence vectors on the basis of HLSTM.

14) HIERARCHICAL LSTM WITH SENTENCE-LEVEL LDA ATTENTION (HLSTM-SA-LDA)

We add Sentence-level LDA Attention into the second stage for composing document representation on the basis of HLSTM.

15) TREE-LSTM + CHAIN-LSTM (TCLSTM)

It is a hierarchical structure of two neural networks: The lower layer is a Tree-LSTM network to compute sentence representations; and the upper layer is a Chain-LSTM network to compute the document representation. No attention mechanism is used here.

16) TREE-LSTM WITH WORD-LEVEL ATTENTION + CHAIN-LSTM (TCLSTM-WA)

We include a general attention mechanism based on MLP [27] into the lower layer of the above TCLSTM scheme when composing sentence vectors.

17) TREE-LSTM + CHAIN-LSTM WITH SENTENCE-LEVEL ATTENTION (TCLSTM-SA)

We include a general standard attention mechanism based on MLP [27] into the upper layer of the above TCLSTM structure when composing a document representation.

18) TREE-LSTM WITH WORD-LEVEL LDA ATTENTION + CHAIN-LSTM (TCLSTM-WA-LDA)(PROPOSED)

This is our proposed scheme for only single-sentence documents. Notice that in the dataset **SemEval** with only single-sentence documents, the sentence representation is also used as the document representation.

19) TREE-LSTM + CHAIN-LSTM WITH SENTENCE-LEVEL LDA ATTENTION (TCLSTM-SA-LDA)(PROPOSED)

This is our proposed scheme for multi-sentence documents or a mixture of multi-sentence documents and single-sentence documents.

E. EXPERIMENTAL RESULTS

Table 3 summarizes the experiment results for the two evaluation metrics Micro-F1 and AP.

We first discuss the performance of the schemes proposed in the literature. To this end, we divide these schemes into three groups: the word-emotion models that predict social emotions based on word features (including SWAT, ET), the topic-emotion models that associate topic with social emotions (including ETM, WMCM, CSTM) and the neural network-based models (including 1-HNN-BTM, Weighted PCNN, CNN, CNN-SVM). Comparing the word-emotion model with the topic-emotion model, the latter performs better on the **SinaNews** and **SemEval** dataset, and also achieves competitive results on the **ISEAR** dataset. This may be attributed to that leveraging topic features can distinguish between different emotions expressed by a same word in different contexts. As for neural network-based models, they outperform the other two groups of schemes on the **SemEval** and **ISEAR** dataset with their ability of learning semantic features of a document. But the results need to be improved on the **SinaNews** dataset. One possible reason is that the documents in the **SinaNews** dataset are much longer sequences, each of which is composed of many sentences. These neural network-based models (i.e., CNN, CNN-SVM) treat each document as a single sequence without distinguishing the intra-sentence and inter-sentence features, which might lead to a low quality of the learned document representation.

Fig. 4 compares the performance of our proposed models with that of the best HLSTM model and the best scheme

TABLE 3. Experimental results on the three datasets.

	Models	SinaNews		SemEval		ISEAR	
		MicroF1	AP	MicroF1	AP	MicroF1	AP
Schemes from literature	SWAT [6]	38.97	0.40	36.99	-	26.29	0.21
	ET [1]	48.04	0.43	41.06	-	51.12	0.38
	ETM [1]	54.19	0.49	35.44	-	48.79	0.35
	WMCM [11]	-	-	41.71	-	-	-
	CSTM [9]	40.74	0.43	-	-	28.23	0.19
	1-HNN-BTM [29]	-	-	-	-	51.21	0.40
	Weighted PCNN [18]	-	-	51.44	-	-	-
	CNN [37]	51.23	-	-	-	-	-
HLSTM Models	CNN-SVM [38]	52.63	-	-	-	-	-
	SOM [33]	58.59	0.64	-	-	-	-
	HLSTM	61.69	0.68	47.56	0.46	57.60	0.54
	HLSTM-WA	62.10	0.68	48.37	0.47	59.39	0.56
	HLSTM-SA	61.73	0.68	-	-	58.12	0.54
	HLSTM-WA-LDA	63.73	0.68	48.37	0.45	60.53	0.57
TCLSTM Models	HLSTM-SA-LDA	63.31	0.69	-	-	57.96	0.54
	TCLSTM	62.57	0.68	51.22	0.46	59.65	0.57
	TCLSTM-WA	63.36	0.70	50.00	0.44	60.44	0.57
	TCLSTM-SA	62.99	0.69	-	-	60.18	0.56
	TCLSTM-WA-LDA (proposed)	63.64	0.68	52.44	0.47	59.49	0.56
	TCLSTM-SA-LDA (proposed)	65.50	0.71	-	-	60.57	0.57

in the literature. In the SSDoc **SemEval** dataset, the proposed TCLSTM-WA-LDA improves *MicroF1* by 1.00% over the best scheme in the literature. In the MSDoc **SinaNews** dataset, the proposed TCLSTM-SA-LDA improves *MicroF1* by 6.91%, and *AP* by 0.07; While in the MSDoc **ISEAR** dataset, it improves *MicroF1* by 9.36%, *AP* by 0.17 over the best scheme in the literature. Notice that those HLSTM models are not from the literature. Instead, we design these HLSTM models in experiments for comprehensive comparisons. Although they are not our focus in this paper, we notice that in most cases, these HLSTM models perform better than the best scheme in the literature.

On one hand, we attribute the improvements to the adoption of a hierarchical network structure. The lower layer network encodes word-to-word relations into a sentence representation, and the upper network can learn inter-sentence features when composing a document representation. Meanwhile, another difference from the existing neural network-based schemes is that we implement the lower layer with a tree-structured network (i.e., Tree-LSTM), which incorporates the syntactic information into the learning of sentence representations, further improving the ability to learn long-distance dependencies in between words. In Table 3, the one-to-one comparison between HLSTM and TCLSTM, that is, between HLSTM-{WA, SA, WA-LDA, SA-LDA} and TCLSTM-{WA, SA, WA-LDA, SA-LDA}, demonstrates that the TCLSTM models generally perform better than those HLSTM models, which verifies the effectiveness of including the syntactic information when learning a document representation.

On the other hand, like the topic-emotion schemes in the literature, our proposed scheme also values the topic information in social emotion classification. With the help of the designed LDA attention mechanism, the words or sentences that convey more topic information of the input document are

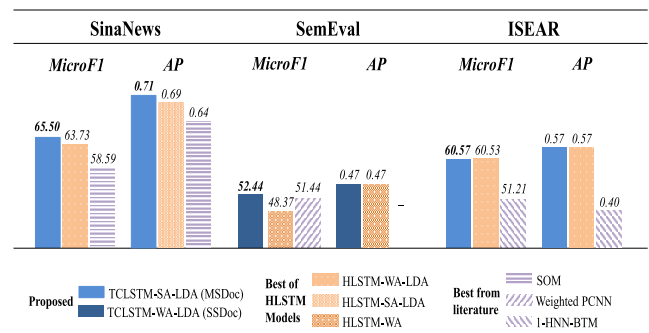


FIGURE 4. Performance comparison among our proposed models, the best HLSTM Model and the best scheme in the literature for the three datasets.

assigned more attentions in the final document representation. In Table 3, compared to the models without attention (such as TCLSTM), the models with the LDA attention (such as TCLSTM-{WA-LDA, SA-LDA}) generally achieve better results. Furthermore, by one-to-one comparing TCLSTM-{WA, SA}-LDA with TCLSTM-{WA, SA}, we can find that the models with the LDA attention also outperform the models with the MLP-based attention mechanism. This can also be observed among those HLSTM models. One possible reason is that the MLP-based attention cannot well capture the key words or sentences for social emotion classification, yet the proposed LDA attention can.

F. ILLUSTRATION OF LDA ATTENTION

Fig. 5 illustrates the computation of the word-level and sentence-level LDA attention with a multi-sentence document from the **ISEAR** experiments. After having trained a LDA model, we can obtain the topic distribution of the document, the topic distribution of each word, and the word

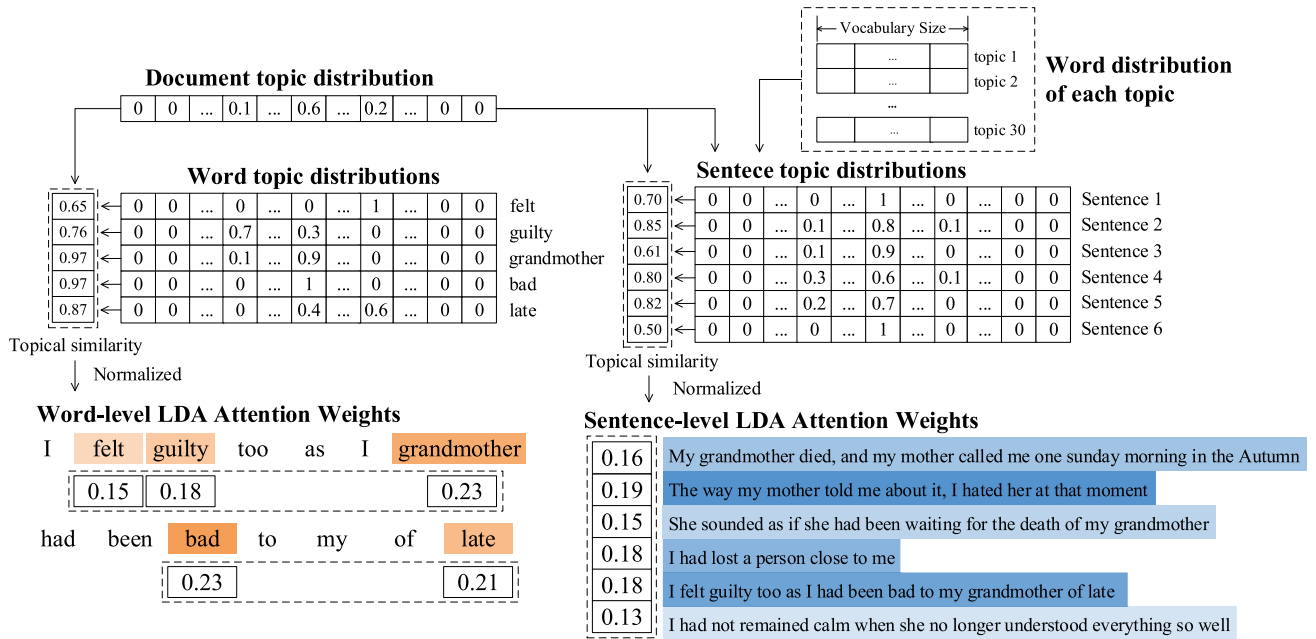


FIGURE 5. Illustration of LDA attention. After having trained a LDA model, we can obtain the topic distribution of the document, the topic distribution of each word, and the word distribution of each topic. The left part illustrates the computation of word-level LDA attention for the 5th sentence. The topic similarity between each word of a sentence and the document is calculated by (1), and the word-level LDA attention is then obtained by (2). Note that the stop words, like “I”, “to”, “as”, “had”, “been”, “to”, “my”, “of” are not considered in the LDA model, so they have no topic distribution. The right part illustrates the computation of sentence-level LDA attention. The topic probability distribution of one sentence is computed by (3), and the topical similarity between a sentence and the document is computed by (5). Finally, the sentence-level LDA attention weights are computed by (6). The bottom of the figure marks the attention weights in different colors according to their values.

distribution of each topic. The left part of the figure illustrates the computation word-level LDA attention for the 5th sentence in the document. We can find the word “bad”, “late”, “guilty” which can trigger a “sadness” emotion are assigned with higher weights; While the neutral word “felt” is assigned a smaller weight. The right part of the figure illustrates the computation of sentence LDA attention. We can observe that the 2nd, 4th, and 5th sentence get greater weights. A close look reveals that they contain obvious words related to the emotion of the document “sadness” like “hated”, “guilty”, “lost”. In contrast, as no emotional words present in the last sentence, it gets a smaller attention.

V. CONCLUSION

In this paper, we have proposed hierarchical tree-structured neural networks with LDA attention for social emotion classification. The lower layer Tree-LSTM network encodes sentence syntactic information to sentence representation according to the sentence dependency tree analysis. We have also proposed to include LDA attention to identify document topic-related key words or key sentences and assign them higher weights when composing document representation. Experiments on three public datasets show that the proposed schemes with integration of syntactic and topical information can effectively improve system performance in terms of higher *MicroF1* and *AP* in both SSDoc and MSDoc dataset, compared with the state-of-the-art schemes.

In this paper, we have mainly focused on mining the syntactic information within a sentence. In our future work, we would like to further mine some hidden relations in between sentences, say for example, by exploring some linguistic knowledge when modeling a paragraph and a document.

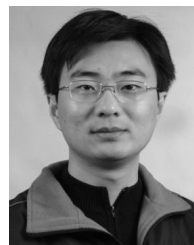
REFERENCES

- [1] S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, and Y. Yu, “Mining social emotions from affective text,” *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 9, pp. 1658–1670, Sep. 2012.
- [2] J. Lei, Y. Rao, Q. Li, X. Quan, and L. Wenyin, “Towards building a social emotion detection system for online news,” *Future Gener. Comput. Syst.*, vol. 37, pp. 438–448, Jul. 2014.
- [3] D. Jiang, X. Luo, J. Xuan, and Z. Xu, “Sentiment computing for the news event based on the social media big data,” *IEEE Access*, vol. 5, pp. 2373–2382, 2017.
- [4] S. Aloufi and A. El Saddik, “Sentiment identification in football-specific tweets,” *IEEE Access*, vol. 6, pp. 78609–78621, 2018.
- [5] C. Strapparava and R. Mihalcea, “SemEval-2007 task 14: Affective text,” in *Proc. 4th Int. Workshop Semantic Eval.* Stroudsburg, PA, USA: Association for Computational Linguistics, 2007, pp. 70–74.
- [6] P. Katz, M. Singleton, and R. Wicentowski, “SWAT-MP: The semeval-2007 systems for task 5 and task 14,” in *Proc. 4th Int. Workshop Semantic Eval.* Stroudsburg, PA, USA: Association for Computational Linguistics, 2007, pp. 308–313.
- [7] Y. Rao, Q. Li, X. Mao, and L. Wenyin, “Sentiment topic models for social emotion mining,” *Inf. Sci.*, vol. 266, pp. 90–100, May 2014.
- [8] Y. Rao, Q. Li, L. Wenyin, Q. Wu, and X. Quan, “Affective topic model for social emotion detection,” *Neural Netw.*, vol. 58, pp. 29–37, Oct. 2014.
- [9] Y. Rao, “Contextual sentiment topic model for adaptive social emotion classification,” *IEEE Intell. Syst.*, vol. 31, no. 1, pp. 41–47, Jan./Feb. 2016.

- [10] Y. Rao, H. Xie, J. Li, F. Jin, F. L. Wang, and Q. Li, "Social emotion classification of short text via topic-level maximum entropy model," *Inf. Manage.*, vol. 53, no. 8, pp. 978–986, Dec. 2016.
- [11] X. Li, H. Xie, Y. Rao, Y. Chen, X. Liu, H. Huang, and F. Lee Wang, "Weighted multi-label classification model for sentiment analysis of online news," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Jan. 2016, pp. 215–222.
- [12] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
- [13] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*. [Online]. Available: <https://arxiv.org/abs/1301.3781>
- [14] 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.
- [15] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*. [Online]. Available: <https://arxiv.org/abs/1408.5882>
- [16] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1422–1432.
- [17] X. Zhao, C. Wang, Z. Yang, Y. Zhang, and X. Yuan, "Online news emotion prediction with bidirectional lstm," in *Proc. Int. Conf. Web-Age Inf. Manage.*, Nanchang, China. Cham, Switzerland: Springer, 2016, pp. 238–250.
- [18] X. Li, Y. Rao, H. Xie, X. Liu, T.-L. Wong, and F. Lee Wang, "Social emotion classification based on noise-aware training," *Data Knowl. Eng.*, to be published.
- [19] R. Socher, B. Huval, C. D. Manning, and A. Y. Ng, "Semantic compositionality through recursive matrix-vector spaces," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.* Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 1201–1211.
- [20] R. Socher, J. Bauer, A. Y. Ng, and C. D. Manning, "Parsing with compositional vector grammars," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, Aug. 2013, pp. 455–465.
- [21] R. Socher, D. Chen, C. D. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 926–934.
- [22] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," 2015, *arXiv:1503.00075*. [Online]. Available: <https://arxiv.org/abs/1503.00075>
- [23] M. Huang, Q. Qian, and X. Zhu, "Encoding syntactic knowledge in neural networks for sentiment classification," *Trans. Inf. Syst.*, vol. 35, no. 3, p. 26, 2017.
- [24] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2204–2212.
- [25] W. Yin, H. Schütze, B. Xiang, and B. Zhou, "ABCNN: Attention-based convolutional neural network for modeling sentence pairs," 2015, *arXiv:1512.05193*. [Online]. Available: <https://arxiv.org/abs/1512.05193>
- [26] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: <https://arxiv.org/abs/1409.0473>
- [27] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2016, pp. 1480–1489.
- [28] F. Kokkinos and A. Potamianos, "Structural attention neural networks for improved sentiment analysis," 2017, *arXiv:1701.01811*. [Online]. Available: <https://arxiv.org/abs/1701.01811>
- [29] X. Li, Y. Rao, H. Xie, R. Y. K. Lau, J. Yin, and F. L. Wang, "Bootstrapping social emotion classification with semantically rich hybrid neural networks," *IEEE Trans. Affective Comput.*, vol. 8, no. 4, pp. 428–442, Oct./Dec. 2017.
- [30] D. Klein and C. D. Manning, "Accurate unlexicalized parsing," in *Proc. 41st Annu. Meeting Assoc. Comput. Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, vol. 1, 2003, pp. 423–430.
- [31] W. Che, Z. Li, and T. Liu, "LTP: A chinese language technology platform," in *Proc. 23rd Int. Conf. Comput. Linguistics, Demonstrations*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 13–16.
- [32] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Jan. 2003.
- [33] X. Li, Q. Peng, Z. Sun, L. Chai, and Y. Wang, "Predicting social emotions from readers' perspective," *IEEE Trans. Affective Comput.*, vol. 10, no. 2, pp. 255–264, Apr./Jun. 2017.
- [34] K. R. Scherer and H. G. Wallbott, "Evidence for universality and cultural variation of differential emotion response patterning," *J. Personality Social Psychol.*, vol. 66, no. 2, pp. 310–328, 1994.
- [35] R. Kulkarni, "A million news headlines," Harvard Dataverse, V3, 2018. doi: [10.7910/DVN/SYBGZL](https://doi.org/10.7910/DVN/SYBGZL).
- [36] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng, "Cheap and fast—But is it good? Evaluating non-expert annotations for natural language tasks," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Honolulu, HI, USA. Stroudsburg, PA, USA: Association for Computational Linguistics, Oct. 2008, pp. 254–263.
- [37] S. Poria, E. Cambria, and A. Gelbukh, "Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 2539–2544.
- [38] S. Poria, E. Cambria, D. Hazarika, and P. Vij, "A deeper look into sarcastic tweets using deep convolutional neural networks," 2016, *arXiv:1610.08815*. [Online]. Available: <https://arxiv.org/abs/1610.08815>



CHANG WANG received the B.S. degree from the School of Electronic Information and Communications, Huazhong University of Science and Technology (HUST), Wuhan, China, in 2018, where he is currently pursuing the master's degree. His research interests include neural networks and sentiment analysis.



BANG WANG received the B.S. and M.S. degrees from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1996 and 2000, respectively, and the Ph.D. degree from the Electrical and Computer Engineering (ECE) Department, National University of Singapore (NUS), in 2004. He is currently a Professor with the School of Electronic Information and Communications, HUST. His research interests include machine intelligence, network science, and social computing technologies.



MINGHUA XU received the B.A. degree from the School of Journalism and Information Communication, Huazhong University of Science and Technology (HUST), in 2002, and the M.A. and Ph.D. degrees from the Social Science Department, National University of Singapore (NUS), in 2005 and 2010, respectively. She is currently an Associate Professor with the School of Journalism and Information Communication, HUST. Her research interests mainly include theory and practice of information communication, recommendation systems, and emotion analysis.

...