World Wide Web (2021) 24:2019–2048
https://doi.org/10.1007/s11280-021-00954-8

SINN: A speaker influence aware neural network model for emotion detection in conversations

Shi Feng[1] · Jia Wei[1] · Daling Wang[1] · Xiaocui Yang[1] · Zhenfei Yang[1] · Yifei Zhang[1] · Ge Yu[1]

Received: 12 September 2020 / Revised: 20 July 2021 / Accepted: 13 September 2021 /
Published online: 6 October 2021

Abstract

Inferring the sentiment polarity or emotion category of subjective text is the fundamental task of sentiment analysis. Recently, emotion detection in conversations that considering context utterances has emerged as a very important and challenging task in this line of research. Most existing studies do not distinguish different speakers in a dialog and fail to characterize inter-speaker dependencies for emotion detection. In this paper, we propose a **S**peaker **I**nfluence aware **N**eural **N**etwork model (dubbed as SINN) to predict the emotion of the last utterance in a conversation, which explicitly models the self and inter-speaker influences of historical utterances with GRUs (Gated Recurrent Units) and hierarchical attention matching network. Moreover, the empathy phenomenon is also considered by an emotion state tracking component in SINN. Finally, the target utterance representation is enhanced by speaker influence aware context modeling, where an attention mechanism is used to extract the most relevant features for emotion classification. We construct a large-scale multi-turn Chinese dialog dataset WBEmoDialog, where each utterance is manually annotated with an emotion label. Extensive experiments are conducted on public available DailyDialog dataset as well as our constructed WBEmoDialog dataset, and the results show that our model can achieve better or comparable performance with the strong baseline methods.

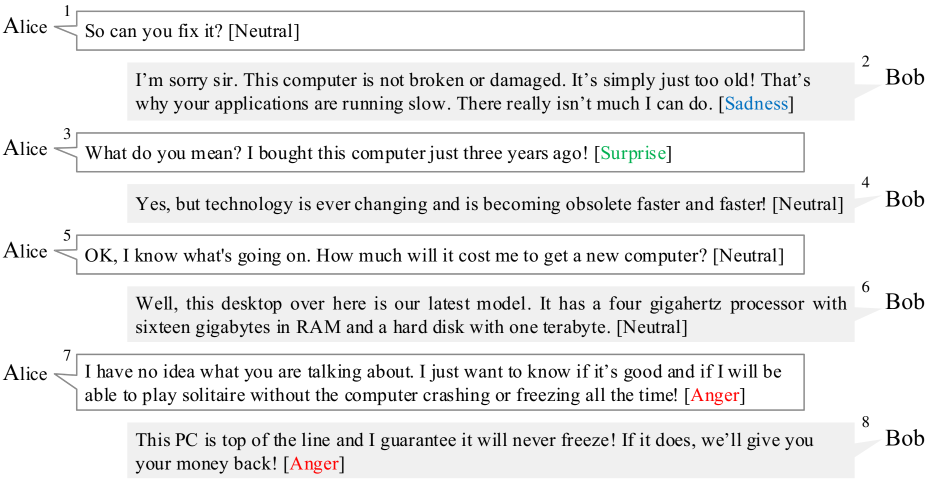
Keywords Conversational emotion detection · Self-influence · Inter-speaker influence · Attention model

1 Introduction

Inferring the sentiment polarity (e.g. positive or negative) or emotion category (e.g. Love, Fear, or Disgust) of subjective text is the fundamental task of sentiment analysis. Existing methods focus on automatically building sentiment lexicons [34, 41] or leverage machine learning models to recognize embedded emotions in the sentences [2, 8]. Despite the

✉ Shi Feng
fengshi@cse.neu.edu.cn

Extended author information available on the last page of the article.



Alice 1: So can you fix it? [Neutral]

Bob 2: I'm sorry sir. This computer is not broken or damaged. It's simply just too old! That's why your applications are running slow. There really isn't much I can do. [Sadness]

Alice 3: What do you mean? I bought this computer just three years ago! [Surprise]

Bob 4: Yes, but technology is ever changing and is becoming obsolete faster and faster! [Neutral]

Alice 5: OK, I know what's going on. How much will it cost me to get a new computer? [Neutral]

Bob 6: Well, this desktop over here is our latest model. It has a four gigahertz processor with sixteen gigabytes in RAM and a hard disk with one terabyte. [Neutral]

Alice 7: I have no idea what you are talking about. I just want to know if it's good and if I will be able to play solitaire without the computer crashing or freezing all the time! [Anger]

Bob 8: This PC is top of the line and I guarantee it will never freeze! If it does, we'll give you your money back! [Anger]

Figure 1 An example conversation from the DailyDialog dataset [25]

relatively ample literature on emotion classification in text, recognizing the emotion in conversations, which relies on not only the target utterance but also the context information, still needs further exploration.

With the explosive growth of social media, massive conversations are produced through online platforms (e.g. WeChat,[1] Twitter,[2] and Weibo[3]) in the Internet every day. Conversational emotion detection is expected to play a pivotal role in many applications such as cyber-crime investigation [21], human-robot emotional interaction [39, 54, 55], customer service [40], and mental health support [1, 11, 18, 44]. Moreover, the context-dependent characteristics of sentences in the conversations have brought in complex challenges for the emotion detection task. Thus, how to effectively make use of context information and detect emotions in conversations have gained massive attention amongst the communities of artificial intelligence and natural language processing researchers.

A conversation consists of a sequence of utterances (two utterances at least) and each utterance is produced by a participant (the speaker). It is generally known that the emotional dynamics in conversations are driven by two factors: self and inter-speaker emotional influence [31]. Self-influence reflects the speakers' own willingness to keep or change their emotions during dialogue. That means the emotion of the current utterance is closely related to the emotions of the speakers' past utterances. On the other hand, inter-speaker influence relates to emotional dynamics induced by the counterparts in the dialogue.

Take Figure 1 from the DailyDialog dataset [25] as an example, we can see that each speaker tends to keep her/his own emotional state, while the speaker's emotion can also be influenced by the other speaker involved. Alice's emotion changes according to what Bob has said (3rd, 5th, 7th utterances), which reflects an interactive inter-speaker influence of Bob on Alice. This interactive influence is mutual since Bob's emotional state also depends on Alice. At the end of the dialogue, Alice turns into anger because her question has not been solved yet, and this enrages Bob to make an angry response, reflecting a kind of emotional

[1] https://web.wechat.com/

[2] https://www.twitter.com/

[3] https://www.weibo.com/

infection of inter-speaker influence because that one may transfer his/her emotion to others easily as a result of empathy.

Despite the complex interactive emotional states of speakers in dialogue, most of the previous literature do not distinguish different speakers in a conversation and treat the context utterances only as a textual sequence. Recently, Hazarika et al. [16] proposed a Conversational Memory Network (CMN) model to feed speaks' historical utterances into memory network, where each speaker is associated with separate memory cell. Following this idea, Hazarika et al. [15] further utilized Gated Recurrent Unit (GRU) [5] to model the influence between speakers. Although these methods have achieved promising results, the inter-speaker influences are modeled by linear GRU utterance sequence or memory network, which could not fully capture the dependencies between the speakers during the dialogue.

To tackle these challenges, in this paper we propose a **S**peaker **I**nfluence aware **N**eural **N**etwork model (dubbed as SINN) for emotion detection in conversations, which models the self and inter-speaker emotional influences explicitly and comprehensively. SINN first adopts GRUs to deal with historical utterances of the target utterance based on each speaker and these histories are fed into two separate sections, which will extract speakers' interactive emotional features and track empathic states simultaneously. After that, the interactions between self as well as inter-speaker influence features with the target utterance are calculated by attention mechanism to synthesize important contextual features. Finally, the target utterance and the weighted contextual features by attentions are concatenated as a final representation which is used to predict the emotion category on the target.

The conversational emotion detection research also suffers from the shortage of large-scale annotated datasets. Some existing dialog datasets are small in size [3, 33] and are not appropriate for complex neural network models. The CSSA Chinese dialog dataset is labeled only at the dialog level, which is too coarse to reflect the emotion change in the conversation [56]. In this paper, we manually construct a large-scale Chinese dialog dataset WBEmoDialog, which has more than 10,000 dialogs and 45,000 utterances. Each utterance is annotated with an emotion label, and thus WBEmoDialog becomes an appropriate benchmark dataset for the conversational emotion detection task.

To sum up, the main contributions of this paper are as follows.

- We propose a novel framework called SINN to detect emotions in conversations. SINN leverages two GRUs to model speakers' self influence separately and utilizes a hierarchical matching network to comprehensively model the inter-speaker influence.
- We develop a large-scale Chinese dialog dataset WBEmoDialog,[4] where each utterance is manually annotated with an emotion label.
- Extensive experiments are conducted on public available DailyDialog dataset and our constructed WBEmoDialog dataset, and the results show that our model can achieve better or comparable performance with the strong baseline methods.

2 Related work

2.1 Emotion analysis

Detecting the emotion of text in social media is the fundamental task of sentiment analysis [43, 45]. The existing studies can be generally categorized into lexicon-based [35, 41] and

[4] WBEmoDialog dataset is released at https://github.com/YangXiaocui1215/WBEmoDialog

learning-based [2, 10] methods. Yang et al. manually constructed an emotion ontology to recognize emotions in microblogs [13]. Wen et al. proposed a lexicon and learning hybrid based method for tweet emotion prediction [49].

The lexicon-based or traditional learning based emotion classification methods rely on manually selected features or emotion lexicons that are really labor-intensive. Recently, we have witnessed the rise of deep learning methods for emotion classification, which can automatically learn discriminative features from a large dataset. Feng et al. proposed a ranking based convolutional neural network model for multi-label emotion detection in Chinese microblogs [10]. Yang et al. leveraged the event-driven attention model to rank the emotional reactions when people reading online news articles [50]. Zhang et al. introduced the emotion distribution learning problem and proposed a multi-task convolutional neural network for conducting text emotion distribution prediction and classification simultaneously [53]. Although the existing methods have achieved promising results, most of these models regard each text as an independent training instance and ignore the context information such as the previous utterances in the conversation.

Recently, researchers have leveraged emotion analysis techniques for psychological counseling and mental health support. Liu et al. manually constructed an emotion dataset grounded on the Helping Skills Theory [17] for emotional support [27]. Focusing on the same issue, Sun et al. built a Chinese dataset of psychological health support in the form of question and answer pair [42]. The answer strategy classification method with typical lexical features and answer generation models were evaluated on the new datasets. To tackle the low sentiment resource problem in the healthcare area, Liu et al. proposed a new feature extraction approach using position embeddings to generate a medical sentiment lexicon for drug review sentiment analysis [26]. Ferraro et al. employed an SVM model with emotion lexicons as features to classify harmful posts in Internet Support Groups for mental health [11]. Previous studies also leverage feature engineering and machine learning methods to detect autism spectrum disorder in clinical data [18, 44]. These studies have shown the wide applications of emotion analysis techniques.

2.2 Emotion detection in conversations

The previous contextual sentiment analysis studies utilize certain kinds of contextual information in the conversation [20, 37, 46, 47]. Huang et al. proposed a hierarchical LSTM model with two levels of LSTM networks to model the retweeting/replying process and capture the long-range dependencies between a tweet and its contextual tweets [20]. Wang et al. regarded the microblog conversation as sequences and leveraged Bidirectional LSTM to obtain the continuous representation of Chinese microblogs [9, 47]. Ren et al. utilized two sub-modules to study features from conversation-based context, author-based context, and topic-based context about a target tweet, respectively [37]. Vanzo et al. employed a model named SVM^{hmm} using Markovian formulation of the SVM to predict the sentiment polarity of entire sequences of tweets [46]. Zhang et al. built a large-scale human-computer conversation dataset and adopted a single level architecture by using Convolutional Neural Networks (CNNs) for sentiment classification [52]. Gupta et al. proposed a model consisting of two LSTM layers using two different word embedding matrices, Glove and SSWE, for detecting emotions in textual conversations [14]. Luo et al. proposed a self-attentive Bidirectional LSTM (SA-BiLSTM) network, which used self-attention to extract the dependence of all the utterances in the conversation [28].

The main shortage of these methods is that they do not separately treat the speakers in a conversation, namely these models are without awareness of different speakers. Hazarika et al. [16] utilized a Conversational Memory Network (CMN) to amend this shortcoming. CMN considered utterance histories of each speaker to model emotional memories and used memory network to capture inter-speaker dependencies. Then, Hazarika et al. [15] proposed another improved model named Interactive Conversational memory Network (ICON) that incorporated self and inter-speaker influences simultaneously and adopted a multiple hop scheme on them. Our model is inspired by ICON partially while quite different from ICON, where we adopt a more comprehensive approach to model the inter-speaker influences from two aspects, namely interactive dependency as well as empathy.

2.3 Datasets for emotion detection in conversations

Emotion detection in conversations has emerged as a hot research problem in the community. However, only limited conversation datasets with emotion labels have been released for this task. The statistics of six publicly available datasets are listed in Table 1.

EmoContext [4] The original data of this dataset were crawled tweets with replies from Twitter. The preprocessing steps include retweet tag removal, URL replacement, and lowercase converting. About 21% of dialogs in the EmoContext dataset contain emojis. Every dialog has three utterances, and the last utterance is annotated with one of four emotion labels, including *Happy*, *Sad*, *Angry*, and *Others*.

Emotionlines [19] This dataset contains two sub-sets, namely Friends and EmotionPush. Friends are built based on script lines from TV-series *Friends*, and EmotionPush are private Facebook Messenger dialogues. Each sub-set contains 1,000 conversations and every utterance is annotated with one of seven emotion labels, including *Neutral*, *Joy*, *Sadness*, *Fear*, *Anger*, *Surprise*, and *Disgust*.

IEMOCAP [3] This is a multi-modal dialogue dataset including video, speech, motion capture of the face, and text transcriptions. Ten actors were invited to perform selected emotional scripts and also to improvise in hypothetical scenarios designed to elicit specific types of emotions (*Happiness*, *Anger*, *Sadness*, *Frustration*, and *Neutral*).

MELD [33] This is also a multi-modal dialogue dataset, which is built by extending and enhancing the contents of Emotionlines dataset. Each utterance encompasses audio, visual, and textual modalities, and is annotated with the same emotion label set as Emotionlines.

Table 1 The data statistics of the six emotional dialog datasets

Dataset	Dialogues		Utterances		Modality
	Training set	Test set	Training set	Test set	
IEMOCAP	120	31	5810	1623	multi-modal
Emotionlines	800	200	11,739	2,764	text
MELD	1,153	280	11,098	2,610	multi-modal
CCSA	1,730	432	10,130	2,533	text
DailyDialog	12,118	1,000	95,744	7,863	text
EmoContext	32,913	5,508	98,739	16,524	text

DailyDialog [25] This dataset was built by collecting conversations from English learning Websites, thus reflecting our daily way of communication. Each utterance is labeled by one of seven emotion labels, including *Anger*, *Disgust*, *Fear*, *Happiness*, *Sadness*, *Surprise*, and *Neutral*.

CCSA [56] This is a multi-turn Chinese dialogue dataset whose instances are collected from online English learning Websites. The dialogues are labeled with three sentiment polarities and 22 fine-grained emotion classes.

The datasets in Table 1 are ordered by size. IEMOCAP dataset has the least number of dialogs, whereas EmoContext is the largest in size, in which the number of dialogs is far more than the other counterparts. There are some problems with these datasets. The multi-modal IEMOCAP and MELD are too small in size and may suffer from the exaggerated emotion issue because these two datasets are built by actors in hypothetical scenarios. Moreover, the annotators label the datasets by watching the videos, which means the annotators can decide the label of the utterance by visual or audio features instead of text features. Emotionlines also has limited training instances that are not appropriate for capturing emotional features using deep learning models. CSSA is labeled at dialog level, namely the utterances in one dialog share the same emotion label. This is not in line with the actual situation of people's communication, where emotion can change during the conversation. Although EmoContext has a huge number of dialogs, each dialog in this dataset has only three utterances and only the last utterance has an emotion label.

Figure 2 demonstrates the emotion label distributions of the corresponding conversation datasets. We can observe that the label distributions of these datasets are extremely unbalanced. The Neutral/Other label accounts for the overwhelming proportion of each dataset, which is consistent with the laws of our daily conversations.

Among the six datasets, DailyDialog is of a moderate size, which can not only meet the needs of model learning but also has emotion label in every utterance. Therefore, we adopt DailyDialog as the experiment dataset for evaluating the proposed models. In addition, most of the aforementioned datasets are in English, and only CCSA is in Chinese. In this paper, we focus on constructing a large-scale dataset of Chinese conversations (named as WBEmoDialog) to conduct comprehensive experiments for the proposed models. We will elaborate on the construction of a new dataset in Section 5.

3 Model framework

The traditional emotion detection methods treat each instance in the dataset equally and independently, and do not care about the order of instances. However, the embedded emotion of a target utterance highly relies on the preceding utterances in the conversation, as shown in Figure 1.

We propose a new neural network based model to tackle the challenges of speaker influence and interactive dependency in conversations. The emotion detection problem in this paper is defined as follows. Suppose that there are n utterances in a dyadic (two-person) conversation, where the communication between two speakers A and B goes on alternately. Here, a conversation $\mathcal{C} = (u_A^1, u_B^2, u_A^3, u_B^4, ..., u_\lambda^n)$ is ordered temporally, where u_λ^n is the n^{th} utterance spoken by person λ, $\lambda \in \{A, B\}$. Our goal is to predict the emotion of the last utterance in the conversation. The schematic overview of our proposed Speaker Influence aware Neural Network model SINN is shown in Figure 3.

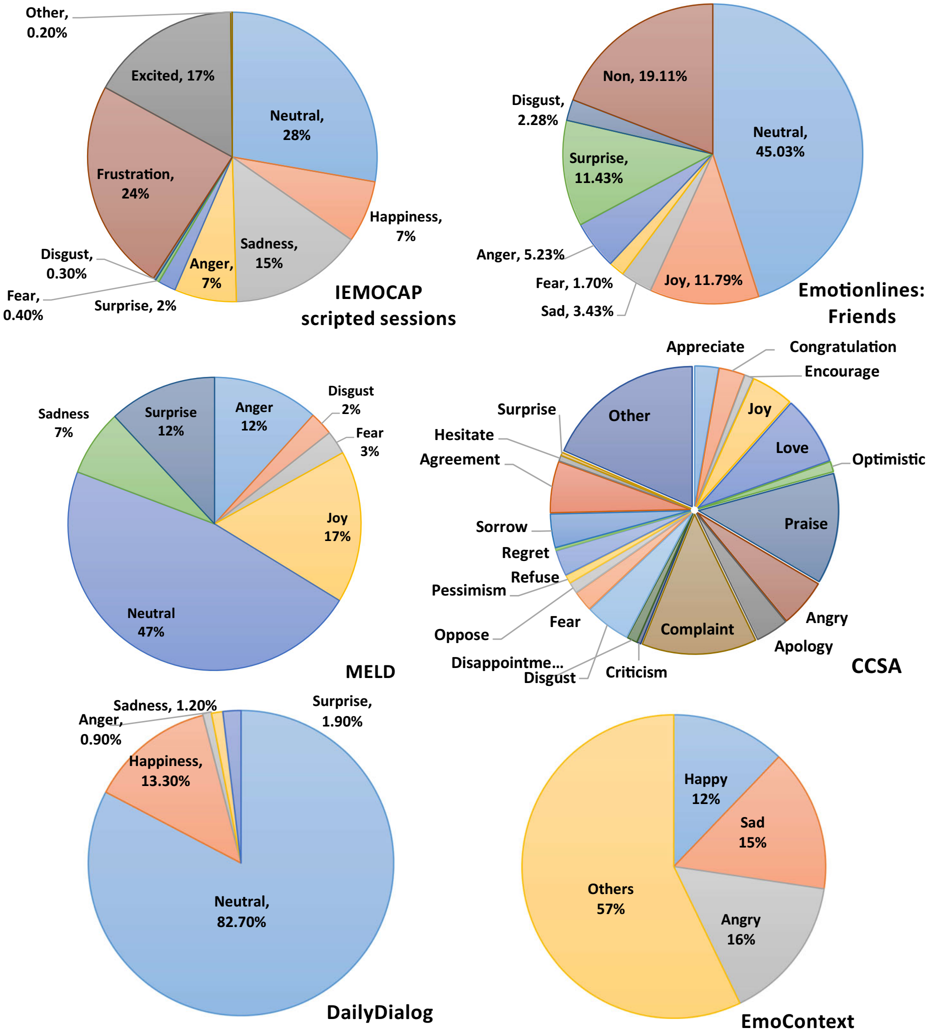


Figure 2 The emotion label distributions of the conversation datasets

For each u_{λ}^{t} in $\mathcal{C}$, $(\forall t \in [1, n])$, we firstly employ Convolutional Neural Networks (CNN) and Gated Recurrent Unit (GRU) on it for feature extraction. The representations $\mathbf{u}$ of an utterance will be a concatenation of the outputs of CNNs and GRUs. To get the contextual clues of the last utterance, the previous $n-1$ utterances are divided by speakers respectively, and separately fed into different GRU_{λ} function to collect self-influences as $\mathbf{H}_{A}$ and $\mathbf{H}_{B}$. Then $\mathbf{H}_{A}$ and $\mathbf{H}_{B}$ will be passed into two components simultaneously, namely interactive dependency matching component and empathy tracking component. These two components can process sequential patterns of historical utterances and incorporate interactive influence and empathic emotional state as inter-speaker influence feature representations $\mathbf{s}$ comprehensively. The enhanced representation $\mathbf{s}$ contains the whole emotional influence and context factors of the n^{th} utterance to be predicted. Due to the fact that each factor in $\mathbf{s}$

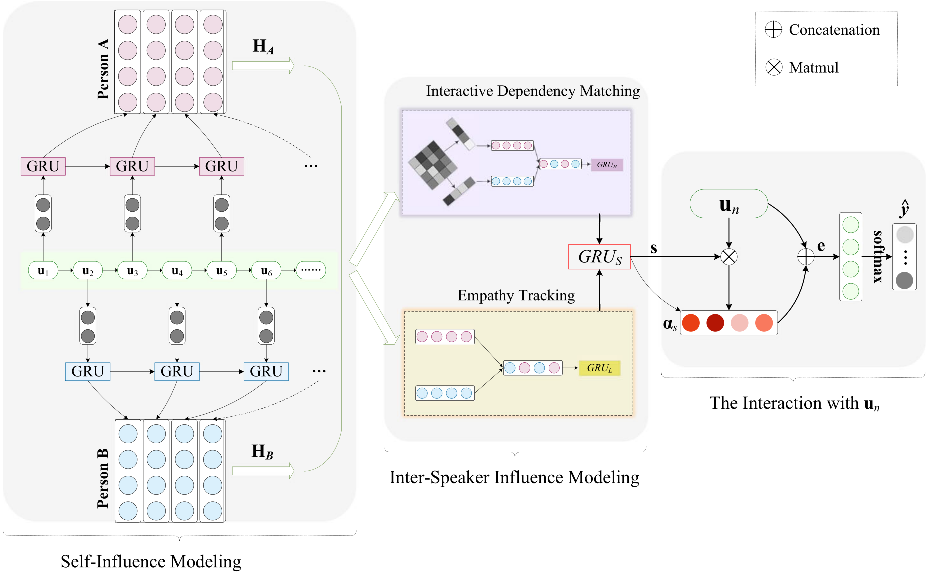


Figure 3 The architecture of Speaker Influence aware Neural Network model SINN. In the self-influence modeling module, $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$... are the learned representations of utterances u_A^1, u_B^2, u_A^3... in the conversation $\mathcal{C}$. Note that we omit the utterance modeling stage by CNN and GRU models in this overall architecture

owns varying degrees of importance, we calculate attention scores α of $\mathbf{s}$ relevant to the target utterance $\mathbf{u}_n$. Finally, we concatenated the weighted $\mathbf{s}$ with utterance $\mathbf{u}_n$ and feed them to a fully connected layer to get the emotion prediction.

As illustrated in Figure 3, our proposed SINN model can be divided into three main parts: (i) self-influence modeling, (ii) inter-speaker influence modeling, and (iii) the interaction with the utterance to be predicted. The second part can be further broken down into two components: (a) interactive dependency matching and (b) empathy tracking. We will elaborate the utterance representation method and the three parts of SINN in the following section.

4 Speaker influence aware neural network

We now describe our method for emotion detection in conversations. We first introduce the CNN and GRU based method to learn the representation of utterances in the conversation. Then, we elaborate the three main parts of the SINN model that captures both the self-influence and inter-speaker influence for enhancing the context representation.

4.1 Utterance modeling

To learn the semantics in the conversation and preprocess the data for further neural models, the first thing to do is to prepare the representation of an utterance. For the n^{th} utterance in the conversation $\mathcal{C}$, pre-trained d-dimensional ELMo embeddings [32] are adopted to represent each word of it. Compared with Word2Vec [29], the ELMo model can adjust the word embeddings according to the context, and thus can leverage more semantic information for better word representation learning.

An utterance with m words is then represented as $u = (\omega_1, \omega_2, ..., \omega_m)$, where ω_i is d-dimensional word embedding based on ELMo for the i^{th} word in the utterance. The embedding vectors are concatenated in word order and we can get a $m \times d$ embedding matrix $\mathbf{W}$ as the original feature representation of u. Then CNN and GRU models are utilized to extract features of matrix $\mathbf{W}$ as shown in Figure 4, and $\mathbf{u}$ is a concatenation of the features from CNN and GRU.

The CNN model employs the sliding filters (covering a number of continuous words) to extract the local features of the utterances. Then the salient features are highlighted by the pooling layer, so as to eliminate the noise data and enhance the model's robustness. In this paper, we utilize TextCNN [23] with 1D convolution kernels to process utterance representation $\mathbf{W}$, as shown in the left part of Figure 4. The filters with the width of 2, 3, 4 are leveraged to conduct the convolution operation, as previous studies have shown that most of the valuable semantic features fall in the size range between 2 and 4 [10]. The extracted local semantic information of n-grams are then fed into the max pooling layer, which captures the most important feature with the highest value for each filter and produces a vector of utterance $\mathbf{u}_{CNN}$ with fixed length.

The TextCNN model can effectively capture the local semantics in various granularities, but fail to learn the long distance dependency between words, as the utterance is

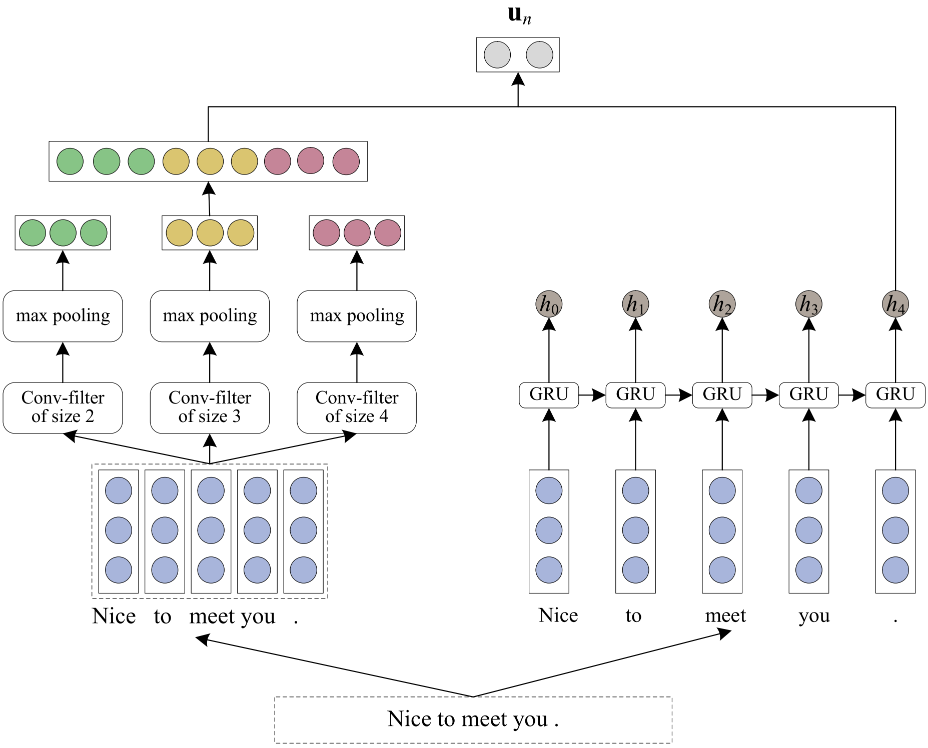


Figure 4 The utterance representation based on CNN and GRU

usually an ordered sequence of many words. As a complement to CNN, in this paper we employ GRU [6] to model the word sequences in utterances. In the right part of Figure 4, the utterance embedding matrix $\mathbf{W}$ is composed of m ordered word vectors, namely $\mathbf{W} = [\omega_1; \omega_2; ...; \omega_m]$. The word vector ω_t in each time step is fed into a GRU module for calculating the hidden vector $h_t = GRU(h_{t-1}, \omega_t)$, where h_{t-1} is the hidden vector of last time step. Thus the input of a GRU module is the embedding vector of a word, and the output hidden vector of the last GRU module can represent the semantics of the whole utterance as well as capture the long distance dependencies in the word sequence. We denote the hidden vector of the last GRU module as $\mathbf{u}_{GRU}$ to represent GRU-based embedding of the utterance.

Finally, the feature vector of $\mathbf{u}_{CNN}$ and $\mathbf{u}_{GRU}$ are concatenated together to get the enhanced utterance representation $\mathbf{u}$, which covers both the local semantics and long distance dependencies in the utterance, and paves the way for the further neural modeling steps.

4.2 Self-influence modeling

Self-influence, also known as *emotional inertia* [24], reflects the phenomenon that speakers tend to keep their emotions unchanged during the conversations, namely the emotional state of the current utterance is usually in correspondence with the speaker's previous ones. Therefore, it is necessary to deal with the utterances made by each speaker separately.

To predict the emotion label of the last utterance in the conversation, we need to model the context information in the speaker's historical utterances. Concretely, since the self-influence only involves speaker himself/herself, we extract different speakers' corresponding historical utterances in the conversation to construct new sequences. Here, for a $\mathcal{C} = (u_A^1, u_B^2, u_A^3, u_B^4, ..., u_\lambda^n)$, we split it into two series according to each speaker, getting $\mathcal{C}_A = (u_A^1, u_A^3, ..., u_A^i)$ for speaker A and $\mathcal{C}_B = (u_B^2, u_B^4, ..., u_B^j)$ for speaker B. $\mathcal{C}_A$ and $\mathcal{C}_B$ are constructed by the utterances in original temporal order, and we define them as new sequence $\mathcal{C}_\lambda = (u_{\lambda,1}, u_{\lambda,2}, ..., u_{\lambda,T})$, where $\lambda \in \{A, B\}$, $i < n$, $j < n$, $T \in \{i, j\}$. The representation of each utterance in $\mathcal{C}_\lambda$ is learned by the CNN and GRU model, as described in Section 4.1. Then for each $\mathcal{C}_\lambda \in \{\mathcal{C}_A, \mathcal{C}_B\}$, we feed them into new GRU model GRU_λ to grasp the temporal history respectively, as shown in Figure 5.

Specifically, in every time step t, the hidden state h_t is calculated as:

$$r_t = \text{sigmoid}\left(\mathbf{W}^r h_{t-1} + \mathbf{U}^r x_t + \mathbf{b}^r\right) \tag{1}$$

$$z_t = \text{sigmoid}\left(\mathbf{W}^z h_{t-1} + \mathbf{U}^z x_t + \mathbf{b}^z\right) \tag{2}$$

$$\tilde{h}_t = \tanh\left(\mathbf{W}^c (h_{t-1} \odot r_t) + \mathbf{U}^c x_t + \mathbf{b}^c\right) \tag{3}$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t \tag{4}$$

where $\mathbf{W}^r$, $\mathbf{U}^r$, $\mathbf{W}^z$, $\mathbf{U}^z$, $\mathbf{W}^c$, $\mathbf{U}^c$ are parameter matrices, $\mathbf{b}^r$, $\mathbf{b}^z$, $\mathbf{b}^c$ are parameter vectors, $\odot$ represents dot product, and x_t denotes the input of time step t, i.e. the utterance modeling vector $\mathbf{u}_t$ ($t \in [1, T]$). GRU_λ adopts speakers' utterances as input, and generate hidden state h of each time step. These hidden state h of all time steps can be concatenated together to form self-influence matrix $\mathbf{H}_\lambda = [h_{\lambda,1}; h_{\lambda,2}; ..., h_{\lambda,T}]$, $\mathbf{H}_\lambda \in \{\mathbf{H}_A, \mathbf{H}_B\}$. $\mathbf{H}_A$ or $\mathbf{H}_B$ represents the historical information of a speaker with his or her own previous utterances. After that, we encode $\mathbf{H}_A$ and $\mathbf{H}_B$ as two matrices to further explore correlations between utterances.

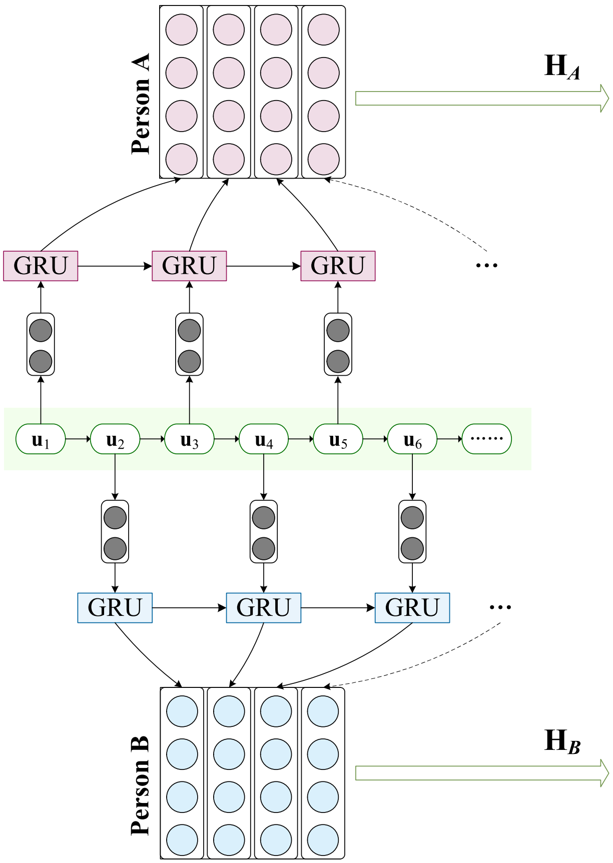


Figure 5 The modeling of self-influence

4.3 Inter-speaker influence modeling

Different from reviews and blogs, the conversation is an interactive way of communication, so the emotional state of one speaker can be affected by the others in the conversation, as shown in Figure 1. This emotional interaction phenomenon is called *inter-speaker influence* [16]. The self-influence only refers to the historical utterances of the speaker himself/herself, but the inter-speaker influence pays more attention to the way of interactions between speakers. Inspired by previous studies [16], we leverage an interactive fusion method for the historical utterances of both speakers to model the inter-speaker influence in the conversation. More concretely, in this subsection two methods called Interactive Dependency Matching and Empathy Tracking are proposed and finally integrated together for this task.

Interactive dependency matching Suppose $\mathbf{H}_A$ and $\mathbf{H}_B$ are self-influence modeling result matrices for historical utterances of speakers A and B. Since utterances constantly interfere with each other, we introduce an interactive mechanism called Interactive Dependency Matching to condense the hidden interplays between them, as shown in Figure 6.

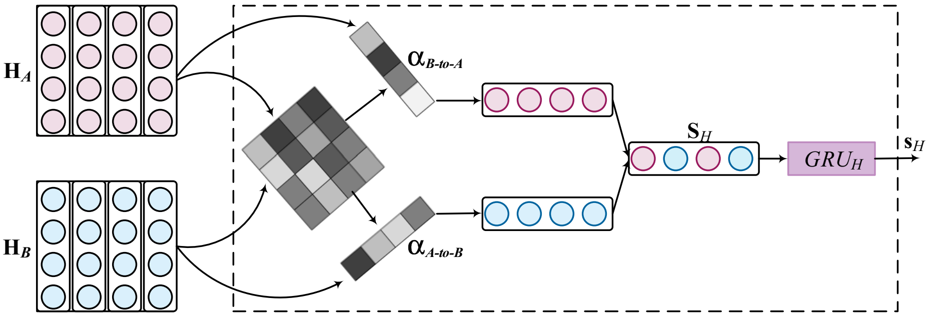


Figure 6 Schematic overview of Interactive Dependency Matching

In Figure 6, different utterances in $\mathbf{H}_A$ and $\mathbf{H}_B$ may have varying influences on the speakers' emotion states. So we first calculate the confusion matrix $\mathbf{H} = \mathbf{H}_A \times \mathbf{H}_B^T$. Given the confusion matrix $\mathbf{H}$, we apply it with attention mechanism [51] from two directions, which could be seen as a $B-$to$-A$ attention and an $A-$to$-B$ attention. The attention mechanism can help us to extract the significant interactive information between $\mathbf{H}_A$ and $\mathbf{H}_B$ and further improve the model's prediction performance. Particularly, we need to calculate the attention scores of both sides involved, $\boldsymbol{\alpha}_{B-to-A}$ (the effect of person B on A) as well as $\boldsymbol{\alpha}_{A-to-B}$ (the effect of person A on B), which are inspired by [38]. Firstly, the attention $\boldsymbol{\alpha}_{B-to-A}$ is calculated as:

$$v_A = \tanh\left(\mathbf{W}_{w_1}\mathbf{H}^{\mathrm{T}} + \mathbf{b}_{w_1}\right) \tag{5}$$

$$\boldsymbol{\alpha}_{B-to-A} = \mathrm{softmax}\left(v_A^{\mathrm{T}}\mathbf{v}_{w_1}\right) \tag{6}$$

$$\mathbf{H}'_A = \mathbf{H}_A\boldsymbol{\alpha}_{B-to-A} \tag{7}$$

where $\mathbf{W}_{w_1}$, $\mathbf{b}_{w_1}$, $\mathbf{v}_{w_1}$ are weight matrix and vectors, and $\boldsymbol{\alpha}_{B-to-A} \in \mathbb{R}^{l_A}$ (l_A is the length of preceding utterances of A) is the attention weight vector implying the influence of person B's utterances on A. More precisely, each element in $\boldsymbol{\alpha}_{B-to-A}$ is the score that represents the importance of each utterance among A's previous utterances. More than that, due to the joining of $\mathbf{H}_B$, which represents the history of B, $\boldsymbol{\alpha}_{B-to-A}$ can also indicates the hidden trails of how B acts on A interactively. After this attention, we get a weighted matrix $\mathbf{H}'_A$ of A's history based on the attention scores $\boldsymbol{\alpha}_{B-to-A}$.

In the same way, we can calculate $\mathbf{H}'_B$ by using the following formulas with different parameters:

$$v_B = \tanh\left(\mathbf{W}_{w_2}\mathbf{H} + \mathbf{b}_{w_2}\right) \tag{8}$$

$$\boldsymbol{\alpha}_{A-to-B} = \mathrm{softmax}\left(v_B^{\mathbf{T}}\mathbf{v}_{w_2}\right) \tag{9}$$

$$\mathbf{H}'_B = \mathbf{H}_B\boldsymbol{\alpha}_{A-to-B} \tag{10}$$

where $\mathbf{W}_{w_2}$, $\mathbf{b}_{w_2}$ and $\mathbf{v}_{w_2}$ are weight matrix and vectors. $\boldsymbol{\alpha}_{A-to-B}$ calculates the attention scores in different direction with $\boldsymbol{\alpha}_{B-to-A}$ as shown in Formula (5) and (8), so different historical utterance information is considered in $\mathbf{H}'_B$.

Then we use Formula (11) to integrate $\mathbf{H}'_A$ and $\mathbf{H}'_B$ into a complete interactive sequence of all previous utterances.

$$\mathbf{S}_H = \left(\mathbf{H}'_{A,1}, \mathbf{H}'_{B,1}, \mathbf{H}'_{A,2}, \mathbf{H}'_{B,2}, ..., \mathbf{H}'_{\lambda,n-1}\right) \tag{11}$$

where $\boldsymbol{H}'_{\lambda,t}$ is the enhanced representation of the t^{th} utterance of speaker λ, and $\mathbf{H}'_{\lambda,n-1}$ is the representation of the utterance adjacent to the target utterance u_n

Intuitively, we recover the original sequences of $\mathcal{C}$ ignoring speakers. $\mathbf{S}_H$ temporally denotes the interdependent weighted abstraction of each utterance. However, for extracting features more effectively, we adopt GRU_H to refine $\mathbf{S}_H$ and the output is viewed as a portion of our inter-speaker influence, which is denoted as $\mathbf{S}_H$.

$$\mathbf{s}_H = GRU_H(\mathbf{S}_H) \tag{12}$$

Empathy tracking In this subsection, we will attempt to model the inter-speaker influence from another perspective. The empathy phenomenon means that the emotional states of the two speakers tend to converge at the end of the conversation [12, 36]. This is also a kind of self-influence as one's utterances can be very contagious, and make the two involving speakers have the same emotion. The empathy tracking module ensures that the model can maintain the empathic trend of $\mathcal{C}$, which will play a great role in inferring the final emotion state. The schematic overview of the empathy tracking module is shown in Figure 7.

For the sake of simplicity, the self-influence modeling result matrices $\mathbf{H}_A$ and $\mathbf{H}_B$ are first aggregated by Formula (13) along the temporal dimension, where the representation of utterance vectors are incorporated with respective emotional labels at the same time.

$$\begin{aligned}\mathbf{S}_L = (&\mathbf{H}_{A,1} \oplus L_{A,1}, \mathbf{H}_{B,1} \oplus L_{B,1}, \mathbf{H}_{A,2} \oplus L_{A,2},\\ &\mathbf{H}_{B,2} \oplus L_{B,2}, \cdots\cdots, \mathbf{H}_{\lambda,n-1} \oplus L_{\lambda,n-1})\end{aligned} \tag{13}$$

where $L_{\lambda,t}$ is the embedding of the utterance's emotion label, and $\oplus$ is the concatenation operation of the vectors.

Similarly, we adopt another GRU_L to refine $\mathbf{S}_L$ to s_L denoting empathic features as another portion of our inter-speaker influence.

$$\mathbf{s}_L = GRU_L(\mathbf{S}_L) \tag{14}$$

Eventually, we have the interactive dependency matching result $\mathbf{s}_H$ and the empathy tracking result $\mathbf{s}_L$ as two complementary inter-speaker influences, which have important contribution to the target emotion prediction. We concatenate the both components to form the final inter-speaker influence features for further progress.

$$\mathbf{s} = GRU_S(\mathbf{s}_H \oplus \mathbf{s}_L) \tag{15}$$

4.4 Sentiment classification based on SINN

Given a conversation $\mathcal{C} = (u_A^1, u_B^2, u_A^3, u_B^4, ..., u_\lambda^n)$, the ultimate goal of SINN is to predict the emotion category of the last utterance u_λ^n, i.e. $\mathbf{u}_n$. Based on the self-influence and inter-Speaker influence modules, our proposed SINN model can enrich the conversation context

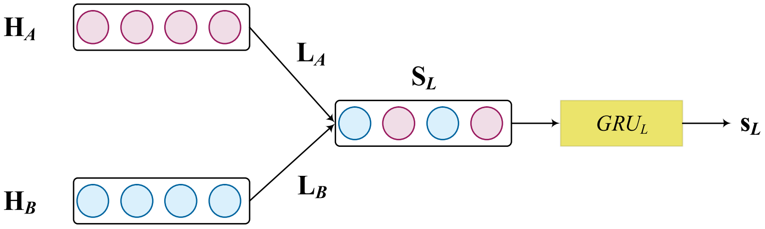


Figure 7 Schematic overview of Empathy Tracking

of $\mathbf{u}_n$ as a speaker influence aware representation $\mathbf{s}$ that considers the sequential information of previous utterances as well as interactive dependencies and empathy states.

In order to capture the attentive dependence of $\mathbf{s}$ relevant to $\mathbf{u}_n$, we perform a mutual calculation between them based on attention mechanism, as shown in the right module of Figure 3 (i.e. the Interaction with $\mathbf{u}_n$). This process can be expressed as

$$\boldsymbol{\alpha}_s = softmax(\mathbf{s}^{\mathrm{T}}\mathbf{u}_n) \tag{16}$$

$$\mathbf{e} = (\boldsymbol{\alpha}_s \odot \mathbf{s}) \oplus \mathbf{u}_n \tag{17}$$

The attention scores $\boldsymbol{\alpha}_s$ with respect to $\mathbf{u}_n$ can assign higher weight to the information relevant to $\mathbf{u}_n$. We update the $\mathbf{s}$ according to $\boldsymbol{\alpha}_s$, and concatenate it with $\mathbf{u}_n$ to be our final emotional representation $\mathbf{e}$. After that, $\mathbf{e}$ is fed into a fully connected layer followed by a softmax layer to predict the emotion of target utterance $\mathbf{u}_n$.

In summary, the learning procedure of SINN model and emotion label classification for dyadic conversations are shown in Algorithm 1.

Algorithm 1 The learning procedure of SINN Model.

```
Input: Dyadic conversation training dataset $D$ (including validation set $V$), testing dataset $T$;
Output: The sentiment classification label set $O$ of the testing dataset;
Preprocess the text in $D$ and $T$, and get the result set $D'$ and $T'$.
Conduct word segmentation for $D'$ and $T'$; Represent each word using pre-trained language model ELMo.
Initialize the parameter set $\Theta$ and learning rate $\alpha$.
repeat
for each mini batch $\in D'$ do
    Use CNN and GRU models to learn the utterance representation $\mathbf{u}$.
    Feed the batch into SINN model, and get the model's prediction labels.
    Calculate the cross entropy loss according to ground-truth labels.
    Use Adam algorithm to update model parameter set, where $\theta = \theta - \alpha\nabla_{\Theta}$.
end for
until Model convergence
Evaluate the performance of the trained model using testing set $T'$, and get the result set $O$.
return $O$
```

4.5 Loss function

In this paper, we regard emotion detection in conversations as a multi-class classification task. The output layer of SINN model has c neurons, which is equal to the number of emotion classes in the dataset. The softmax function in the model's last layer transforms the c dimension values into a probability distribution, and the neuron with the highest probability corresponds to the prediction class of the instance. We leverage the cross-entropy loss function to calculate the differences between the predicted label probability $\hat{y}_i$ and the ground-truth emotion label y_i, as shown in Formula (18).

$$\mathcal{L}(\theta) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{c} y_i log(\hat{y}_i) + \gamma \sum_{\Theta} \theta^2 \tag{18}$$

where N is the number of training instances, γ is the empirically fixed regularization coefficient and θ denotes any trainable parameter in the parameter set Θ. The learning procedure of SINN with the loss function is shown in Line 8 of Algorithm 1.

5 Dataset construction

Several English datasets, such as EmoContext [4], Emotionlines [19], IEMOCAP [3], MELD [33] and DailyDialog [25] have been published for the emotion detection task in the conversations. However, most of these datasets are either small-scaled [3, 19] or built for the multi-modal task [33]. For evaluating the proposed models on the English dataset, we leverage DailyDialog [25] that has more than 10 thousand dyadic dialogues and 80 thousand utterances, where each utterance is associated with one emotion label.

Only limited Chinese conversation datasets have been released for emotion detection task. Zhou et al. built a dialogue corpus CCSA by crawling Chinese Websites for English learning [56]. However, a whole dialogue in CCSA was labeled with only one single emotion. This is a coarse-level annotation, and does not conform to the real situation of emotional interactions, where the transitions of emotion states in utterances are widespread, as shown in Figure 1. In this paper, we focus on constructing a new large-scaled Chinese conversation dataset with emotion labels.

As the most popular Chinese microblogging service, Weibo has more than 500 million registered users and generates more than 100 million posts everyday. Same as Twitter, Weibo allows users to comment on the posts, and thus the 'post-comment' has naturally formed a round of conversation. Since there are a huge number of 'post-comment' pairs, Weibo has become an ideal data source for building conversation dataset.

We do not build the dataset from scratch, but draw support from a public available Weibo conversation dataset ECDT2019,[5] which is originally collected for personalized dialogue generation task. ECDT2019 is a relatively high-quality dataset that has already been preprocessed by hate speech removal and noise data filtering. According to statistics, ECDT2019 has 20.83 million conversations and 56.25 million utterances.

5.1 Data processing

For better adapting to our task, we filter out the dialogues involving more than two speakers in ECDT2019, and conduct the following processing work.

Eliminating the dialogues with languages other than Chinese and English We aim to build a Chinese conversation dataset, but because of the widespread use of English and the code-switch phenomenon, many Chinese conversations are mixed with English words or English names. We have the pre-trained word vectors for both Chinese and English, so we retain the Chinese and Chinese/English code-mixed conversations. The dialogues with other languages are eliminated.

Eliminating short utterances Inevitably, some too short utterances, such as 'Haha!' and 'All right.', will appear in the dialogues. These utterances contain limited information for the context. Thus, we eliminate the dialogues that have utterances within 10 words.

[5] http://conference.cipsc.org.cn/smp2019/evaluation.html

Limiting the dialogue length Due to the characteristics of Weibo, many dialogues only contain two utterances, namely, the post only has one reply. This kind of conversation is too short and makes the problem more like traditional context independent emotion classification. On the other hand, some conversations have too many rounds of utterances, since the reply operation may last for one day or even longer with scattered topics. This is obviously not in line with the normal daily conversations. Therefore, we remove the dialogues with less than four utterances or more than 20 round utterances.

After the aforementioned processing steps, we have about 0.69 million conversations. It is difficult to manually annotate so many data, besides, many of the utterances in this set do not contain emotions. For improving the quality of the dataset, we employ an existing sentiment lexicon to roughly filter out the non-emotional conversations. We calculate the sentiment score of each utterance and eliminate the conversations with lower scores, as shown in Algorithm 2. The intuition of this filtering algorithm is that we can retain high-quality and potentially emotional utterances for further manually labeling steps.

Algorithm 2 Data filtering based on sentiment lexicon.

```
Input: The original Weibo conversation dataset $\mathcal{D}$, the sentiment lexicon $L$;
Output: The filtered dataset $\mathcal{D}'$;
for every conversation $\mathcal{C} \in \mathcal{D}$ do
    for every utterance $u \in \mathcal{C}$ do
        Conduct word segmentation.
        for every word $w_i \in u$ do
            Conduct part-of-speech tagging for $w_i$ and $w_{i-1}$, and calculate the sentiment
            score $s_i$ for $w_i$ based on lexicon $L$.
            if $w_{i-1}$ is a negation word then
                $s_i = s_i \times (-1)$
            end if
            if $w_{i-1}$ is a degree adverb then
                $s_i = s_i \times deg$ and $deg$ is a weight in $L$.
            end if
            $score(u) = \sum_i s_i$
        end for
        $score(\mathcal{C}) = \sum u$
    end for
    if $score(\mathcal{C}) \in (-1, 1)$ then
        Remove $\mathcal{C}$.
    end if
end for
return filtered dataset $\mathcal{D}'$
```

We utilize DLUTE[6] as the sentiment lexicon in Algorithm 2. The lexicon based sentiment analysis method is not so accurate. However, based on the fact that there are a huge amount of data, we can obtain plenty of emotional conversations, which are valuable for building a high-quality dataset.

[6] https://github.com/ZaneMuir/DLUT-Emotionontology

5.2 Data annotation

After data preprocessing and filtering, we get about 10,000 conversations and 45,000 utterances for manual annotation. To facilitate the data labeling task, we implement a multi-user annotation system that can demonstrate the conversations one by one. The data is annotated at utterance level, namely, each utterance is annotated by at least two users with one of the emotion labels {*Happiness*, *Love*, *Sorrow*, *Fear*, *Disgust*, *None*}. The selection of basic emotion labels follows the idea of Paul Ekman [7] but with a minor modification, where some infrequent basic emotions such as *Anger* are removed, and the set is complemented by the label of *None* (no emotion).

Text emotion annotation is a really challenging task, since the labeling results depend on the cognition of each annotator, and different people may have different cognition of emotions. Therefore, almost all the manual annotation tasks for emotions adopt a voting strategy. In addition, the annotators should also stand on the speaker's perspective and comprehensively understand the context of the conversations for achieving better agreement.

In this paper, we invite 5 annotators for labeling the emotions in Weibo conversation dataset. These annotators are graduate students major in sentiment analysis. Firstly, one hundred conversations are sampled from the datasets, and the annotators are trained based on these samples. A discussion is made for an in-depth understanding of the rules and for coordinating the labeling standard. Finally, the annotators start to work on the formal dataset. Every utterance is labeled by at least two annotators. If the first two annotators are consistent, the label of the utterance is finalized. Otherwise, if there is a disagreement, the third annotator will label the data. The final emotion label of the utterance is determined by voting strategy. If all the three annotators have different labels, this conversation is removed from the dataset.

5.3 Data statistics

We denote our annotated Weibo conversation dataset as WBEmoDialog, and the statistics of the dataset are shown in Table 2. We can observe that WBEmoDialog has a relatively large scale and average rounds of 4.4. Based on user habits in Weibo, 4.4 rounds will be the number of replies in a short time with focused topic, so this crawled dataset is able to simulate face-to-face conversations in the real life. According to statistics, 27,167 utterances have consistent labeling results of the first two annotators, accounting for 59.7% of all the utterances. On the other hand, only 600 dialogues are removed, which means the annotators disagree on only a few dialogues. Considering the six-class labeling task, this agreement indicates a dataset with high annotation quality.

Figure 8 demonstrates the distribution of basic emotions in WBEmoDialog. It can be observed that *None* still accounts for a large proportion, which is consistent with the emotion distribution of public available conversation datasets in Figure 2 as well as the real situation of daily life. In addition, WBEmoDialog has more emotional utterances compared with other datasets [19, 33], and the distribution of emotion categories is relatively balanced.

6 Experiments

In this section, we first give a brief introduction to two conversation datasets we use for the emotion detection task. Then we describe our experiment settings and strong baseline

Table 2 The statistics of the WBEmoDialog dataset

Number of dialogues	10,414
Number of utterances	45,498
The max rounds of one dialogue	10
The average round of dialogues	4.4

algorithms for comparisons. Finally, we introduce the empirical results with corresponding discussions. Our experiments are conducted on a commodity PC with NVIDIA 1060 6G GPU.

6.1 Experiment datasets

We conduct experiments on two Datasets: DailyDialog [25] and WBEmoDialog.

6.1.1 DailyDialog

DailyDialog is a high-quality multi-turn dyadic dialog dataset with less noise. The data was collected from English learning Websites for students to practice English conversation in daily life, so the dataset reflects our daily way of communication, and covers a variety of topics. In DailyDialog, each utterance is labeled by one of the seven emotions, including *Anger*, *Disgust*, *Fear*, *Happiness*, *Sadness*, *Surprise* and *Neutral*. The statistics of DailyDialog are shown in Table 3.

In the original DailyDialog dataset, the *Disgust* and *Fear* emotions have limited instances, where *Disgust* and *Fear* account for 0.34% and 0.17% respectively. To alleviate the data imbalance problem, we filter out the instances with *Disgust* and *Fear* emotion labels. Moreover, in order to make the model more fully trained, we chronologically split a dialogue with n utterances into n-1 sub dialogues. For example, for a dialogue $\mathcal{C} = (u_1, u_2, ..., u_m)$, we convert it to $\{(u_1, u_2), (u_1, u_2, u_3), ..., (u_1, u_2, ..., u_m)\}$. So each sub-dialogue includes at least two utterances, i.e. a target utterance has at least one utterance as context. Finally, we obtain the updated DailyDialog dataset for the experiment, as shown in Table 4.

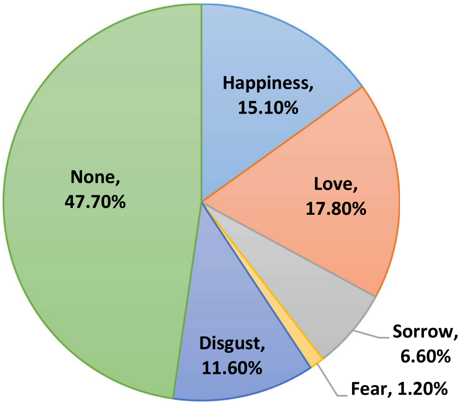


Figure 8 Emotion distributions in WBEmoDialog dataset

Table 3 The statistics of the DailyDialog dataset

Number of dialogues	13,138
The average round of dialogues	7.9
The average number of words in dialogues	114.7
The average number of words in utterances	14.6

Table 4 shows that *Neutral* and *Happiness* occur frequently in the dataset. This is because people usually express no emotion in daily communications or tend to share the joy with each other. The preprocessing steps for DailyDialog are as follows.

- **Punctuation removal**. The non-emotion bearing punctuation marks are removed. But the exclamatory and question marks are preserved as they are potential emotion indicators.
- **Deduplication**. Some characters such as period and blank space are duplicated in the dataset. We eliminate these recurring characters in the utterances.
- **Segmentation**. Some missing blank spaces will generate out of vocabulary words. For example, we could not find the pre-trained embedding for “okay!sure”. So we employ heuristic rules to segment these words to make them reasonable.

6.1.2 WBEmoDialog

WBEmoDialog is the dyadic Chinese text conversation dataset that we build in this paper, and includes six basic emotions. Since each utterance is associated with one emotion label in WBEmoDialog, we also split a dialogue with n utterances into $n-1$ sub dialogues. The final emotion distribution of WBEmoDialog is summarized in Table 5.

We conduct word segmentation for the utterances using Jieba.[7] Note that WBEmoDialog is built based on an exiting conversation dataset, and thus has relatively less noise. We just remove the non-emotion bearing punctuation marks and stop words.

6.2 Experiment setup

To initialize the word embedding matrix, we use the pre-trained 1024-dimension ELMo embeddings [32]. All weight parameters are initialized using the default Tensorflow initializer, and we utilize the Adam optimization algorithm to train them with the learning rate of 0.001. The number of convolutional filters is set as 128 and the filter sizes are set as 2, 3 and 4. The number of GRU cells is 128 for all GRU modules except GRU_S, which contains 256 GRU cells. The weight of L_2 regularization term γ is set 0.001. The dropout rate of 0.5 is set to obtain better performance. The batch size is 128 finally.

For SINN model, cross entropy is adopted as loss function. The original datasets are randomly split into training/validation/testing set with proportion 8:1:1, as shown in Tables 4 and 5. The best model is selected according to the performance on the validation set and we report the model’s performance on the test set. To avoid the overfitting phenomenon, we also adopt an early stop mechanism, i.e. the training will stop when the improvement of the model’s performance on the validation set is less than a certain threshold or when the loss function no longer decreases in a number of batches.

[7] https://github.com/fxsjy/jieba

Table 4 The distribution of each emotion in the updated DailyDialog

Emotion	Training set	Validation set	Testing set	Proportion
Neutral	61,028	6,140	5,248	72,416 (82.7%)
Anger	645	58	92	795 (0.9%)
Happiness	10,113	642	914	11,669 (13.3%)
Sadness	861	65	93	1,019 (1.2%)
Surprise	1,458	96	100	1,654 (1.9%)
Total	74,105	7,001	6,447	87,553 (100%)

6.3 Baselines

We compare our proposed SINN network with the following baseline methods. For a fair comparison, all the word vectors in these baselines are also initialized by ELMo embeddings.

Hierarchical GRU-GRU (**HGG** for short). This baseline model is a two-level GRU network. The first level is a word-level GRU, whose inputs are word vectors of an utterance and each word corresponds to a GRU unit; the hidden vector of the last word represents the embedding of the utterance. The second level is an utterance-level GRU, whose inputs are utterance embeddings of the last layer in chronological order; the last GRU unit corresponds to the target utterance that needs to predict the emotion label, and the output of the last GRU unit is fed into a softmax function for emotion classification. This kind of hierarchical neural network structure has been validated effective for context-aware sentiment classification [9]. Different from [9], HGG adopts GRU instead of LSTM for word and utterance modeling, and does not incorporate the attention mechanism.

Hierarchical CNN-GRU (**HCG** for short). Similar to HGG, HCG adopts a two-level network, while we replace the first level GRU with the CNN model. We implement HGG and HCG models to evaluate different strategies of word and utterance modeling.

Conversational memory network (**CMN** [16] for short). The previous utterances of the two speakers are separately fed into GRU networks for context modeling and generate

Table 5 The distribution of each emotion in the WBEmoDialog

Emotion	Training set	Validation set	Testing set	Proportion
Happiness	4,886	493	552	5,391 (16.9%)
Love	4,518	541	580	5,639 (16.1%)
Sorrow	1,994	202	176	2,372 (6.8%)
Fear	337	34	48	419 (1.2%)
Disgust	3,269	437	461	4,167 (11.9%)
None	13,401	1,632	1,523	16,556 (47.1%)
Total	28,405	3,339	3,340	35,084 (100%)

speaker-aware memory cells. Then CMN reads both the speaker's memories and employs an attention mechanism to find the most useful historical utterances to classify the target utterance.

Interactive conversational memory network (**ICON** [15] for short). As an improved version of CMN, ICON further considers the inter-personal influences between speakers and models these emotional dependencies by memory network with multi-hop attentions.

DialogGCN [22] DialogGCN is a graph neural network based approach for conversational emotion detection, where the self and inter-speaker dependencies between speakers in the context are modeled by a directed graph. The nodes in the graph represent individual utterances and the edges between a pair of nodes/utterances represent the dependency between the speakers of those utterances. Excellent performance has been achieved by DialogGCN on IEMOCAP [3], AVEC [30] and MELD [33] datasets.

6.4 Evaluation metrics

$Precision$ and $F\text{-}Score$ are widely used evaluation metrics for text classification task, and can be calculated as:

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

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

$$F\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{21}$$

where TP, FP, TN and FN mean true positive, false positive, true negative and false negative for the specific emotion class, respectively. We can observe that DailyDialog and WBEmoDialog are unbalanced datasets, i.e. some of the emotion classes have overwhelming instances. Thus we adopt the weighted version of $Precision$ and $F\text{-}Score$ for better evaluating models' performances as did in CMN [16] as well as ICON [15]. More specifically, we calculate $Precision$ and $F\text{-}Score$ for each emotion class and multiply the proportion of each corresponding class for the weighted version metrics.

6.5 Results and discussions

6.5.1 Results on the DailyDialog dataset

The experimental results on the DailyDialog dataset are shown in Tables 6 and 7. As expected, our proposed model SINN, with a novel approach to grasp speaker influence features, obviously outperforms the baseline models HGG, HCG, CMN and ICON. Moreover, SINN also achieves better performance than DialogGCN in macro average $Precision$, macro average $F\text{-}Score$ and weighted average $F\text{-}Score$.

HGG and HCG are hierarchical two-layer neural networks, whose first layer is used for extracting word-level information, and the second layer is used for modeling dialog utterances. In Tables 6 and 7, HCG slightly outperforms HGG in macro average value of $Precision$ (MP for short), and the weighted average value of both $Precision$ (WP for short) and $F\text{-}Score$ (WF for short). However, both HGG and HCG perform worse than the other baselines and SINN. We speculate that the simple two-layer architecture fails

Table 6 The *Precision* comparison results with the baseline models on the DailyDialog dataset

Model	*Neutral*	*Anger*	*Happi.*	*Sadness*	*Surprise*	MP	WP
HGG	0.887	0.383	0.570	0.198	0.269	0.461	0.816
HCG	0.882	0.343	0.584	0.203	0.467	0.496	0.816
CMN	0.883	0.518	0.628	**0.349**	0.398	0.555	0.826
ICON	0.879	**0.533**	0.578	0.276	0.420	0.537	0.816
DialogGCN	**0.968**	0.249	0.489	0.198	0.138	0.417	**0.881**
SINN	0.899	0.490	**0.691**	0.327	**0.470**	**0.575**	0.849

Happi. represents *Happiness*. MP means the macro average of *Precision* across all the emotion classes, and WP means the weighted average of *Precision* across all the emotion classes. The best *Precision* is in bold font, and the second-best *Precision* is underlined

to excavate the deep dependencies between speakers which is extraordinarily important in conversations.

As the improved version of CMN, ICON is a strong baseline model and has been reported to achieve excellent performance in multi-modal conversation datasets [15]. While in this paper, CMN based on our implementation gets a slight advantage over ICON in the two text conversation datasets. Both the ICON and CMN extract each speakers' historical utterances in the conversation separately, and feed them into the different memory networks. The multi-hop attentions are utilized to model the interactions between speakers in conversation. Moreover, both models consider self and inter-speaker influences in a conversation but using a different strategy with SINN. With a similar structure, CMN outperforms ICON by approximately 1% in weighted average *Precision* and *F-Score*.

We can observe that our SINN model can outperform the strong baseline CMN by 2.8% in weighted average *Precision* (WA in Table 6) and can outperform DialogGCN by 9.8% in weighted average *F-Score* (WF in Table 7). This indicates that SINN can capture deeper context information in the historical utterances, and that our self influence modeling method and inter-influence fusion strategy are more effective. In Tables 6 and 7, SINN gets worse performance in some of the *Anger* and *Sadness* categories. These two emotions have limited training instances, namely *Anger* for 0.9% and *Sadness* for 1.2%. Since our SINN model has more parameters, we conjecture that when lack of training data, SINN cannot fully learn

Table 7 The *F-Score* comparison results with the baseline models on the DailyDialog dataset

Model	*Neutral*	*Anger*	*Happi.*	*Sadness*	*Surprise*	MF	WF
HGG	0.900	0.259	0.535	0.184	0.279	0.431	0.819
HCG	0.903	0.289	0.521	0.173	0.350	0.447	0.821
CMN	0.908	0.392	0.525	0.282	0.423	0.506	0.830
ICON	0.902	0.350	0.509	0.249	0.394	0.481	0.821
DialogGCN	0.822	**0.453**	0.589	0.327	0.236	0.485	0.775
SINN	**0.919**	0.350	**0.611**	**0.345**	**0.426**	**0.530**	**0.851**

Happi. represents *Happiness*. MF means the macro average of *F-Score* across all the emotion classes, and WF means the weighted average of *F-Score* across all the emotion classes. The best *F-Score* is in bold font, and the second-best *F-Score* is underlined

Table 8 The *Precision* comparison results with the baseline models on the WBEmoDialog dataset

Model	*Happi.*	*Love*	*Sorrow*	*Fear*	*Disgust*	*None*	MP	WP
HGG	0.787	0.595	0.432	0.00	0.659	0.711	0.531	0.671
HCG	**0.912**	0.620	0.273	0.00	0.485	0.677	0.495	0.648
CMN	0.873	0.637	0.503	0.270	0.644	0.702	0.605	0.694
ICON	0.846	0.601	0.511	0.234	0.645	0.705	0.593	0.687
DialogGCN	0.624	**0.737**	**0.552**	0.476	**0.845**	**0.755**	**0.665**	**0.728**
SINN	0.892	0.636	0.396	**0.600**	0.659	0.733	0.650	0.711

Happi. represents *Happiness*. MP means the macro average of *Precision* across all the emotion classes, and WP means the weighted average of *Precision* across all the emotion classes. The best *Precision* is in bold font, and the second-best *Precision* is underlined

the emotional feature distribution in these two categories, thus can cause poor generalization ability. On the other hand, SINN achieves better performance on *Neural*, *Happiness* and *Suprise* categories in terms of *F-Score*. This phenomenon demonstrates that if given sufficient training data, the SINN model can characterize the features of different emotions and perform much better in the classification task. As the effectiveness and fairness of weighted metrics have been validated in [16] and [15], based on the results of WF, we can conclude that the proposed SINN model has achieved better performance than other baseline methods.

As the reported strong baseline in the literature, DialogGCN fails to achieve the promising results on the DailyDialog dataset in terms of weighted average *F-Score* in Table 6. We conjecture that the performance of DialogGCN is affected by the extremely unbalanced label distribution of the DailyDialog dataset, where DialogGCN pays more attention to the majority emotion *Neural* but neglects the other minority categories.

6.5.2 Results on the WBEmoDialog dataset

The experimental results on WBEmoDialog dataset are shown in Tables 8 and 9.

Firstly, HGG performs better than HCG, but both of the two models fail to detect *Fear* emotion in the dataset, which shows the deficiency of these two-layer models in capturing the rich context information in the conversations. Secondly, CMN slightly outperforms

Table 9 The *F-Score* comparison results with the baseline models on the WBEmoDialog dataset

Model	*Happi.*	*Love*	*Sorrow*	*Fear*	*Disgust*	*None*	MF	WF
HGG	0.822	0.587	0.391	0.00	0.533	0.760	0.516	0.679
HCG	0.834	0.556	0.241	0.00	0.462	0.735	0.471	0.646
CMN	**0.854**	0.617	0.471	0.235	0.436	0.768	0.564	0.687
ICON	0.837	0.611	0.440	0.232	0.442	0.762	0.554	0.679
DialogGCN	0.619	**0.678**	**0.481**	0.290	**0.842**	**0.794**	**0.617**	**0.727**
SINN	0.832	0.593	0.356	**0.353**	0.643	0.779	0.593	0.708

Happi. represents *Happiness*. MF means the macro average of *Precision* across all the emotion classes, and WF means the weighted average of *F-Score* across all the emotion classes. The best *F-Score* is in bold font, and the second-best *F-Score* is underlined

Table 10 The *Precision* comparison results of ablation experiments on the DailyDialog dataset

Model	*Neutral*	*Anger*	*Happi.*	*Sadness*	*Surprise*	MP	WP
w/o ET	0.882	**0.649**	0.662	**0.356**	**0.506**	**0.611**	0.834
w/o IDM	**0.899**	0.295	0.728	0.289	0.481	0.536	0.842
w/o ELMo	0.889	0.000	**0.741**	0.333	0.474	0.488	0.841
SINN	**0.899**	0.490	0.691	0.327	0.470	0.575	**0.849**

Happi. represents *Happiness*. MP means the macro average of *Precision* across all the emotion classes, and WP means the weighted average of *Precision* across all the emotion classes

ICON in MP, MF and WF, which are in line with the results in the DailyDialog dataset. Thirdly, our proposed SINN model outperforms the other baseline methods except DialogGCN in the WBEmoDialog dataset in terms of average metrics over the classes. Besides, our SINN model can achieve relatively good performance for *Fear* emotion where the other models fail. These experimental results demonstrate that the self and inter-speaker influence modeling strategy in SINN is effective and can facilitate the model to capture more valuable information for emotion detection in conversations.

Our SINN model fails to outperform DialogGCN on the average metrics. As the strongest baseline, DialogGCN utilizes both the backward and forward context of the target utterance to build the dialog graph, and thus comprehensively capture the dependency information between speakers. However, the SINN model can only attend the backward context of the target utterance. We conjecture that this is an advantage of DialogGCN over SINN when there are enough training instances (DialogGCN performs much worse on the *Fear* category where the proportion of the emotion accounts for only 1.2% on the WBEmoDialog dataset).

6.6 Ablation experiments

For inter-speaker influence modeling, the SINN model includes two separate components: Interactive Dependency Matching and Empathy Tracking. We conduct the ablation experiments on the two datasets to further evaluate the effectiveness of these two components. The results of the ablation experiments are shown in the Tables 10, 11, 12 and 13.

w/o ET We remove the Empathy Tracking component that including previous emotion labels to evaluate the contribution of this component. On the other hand, the aforementioned baselines (i.e. HGG, HCG, CMN and ICON) do not consider the previous emotion labels of

Table 11 The *F-Score* comparison results of ablation experiments on the DailyDialog dataset

Model	*Neutral*	*Anger*	*Happi.*	*Sadness*	*Surprise*	MF	WF
w/o ET	0.913	**0.372**	0.540	0.313	**0.453**	0.518	0.836
w/o IDM	0.915	0.315	0.562	0.327	0.425	0.509	0.840
w/o ELMo	**0.926**	0.000	**0.611**	0.163	0.416	0.423	0.849
SINN	0.919	0.350	**0.611**	**0.345**	0.426	**0.530**	**0.851**

Happi. represents *Happiness*. MF means the macro average of *F-Score* across all the emotion classes, and WF means the weighted average of *F-Score* across all the emotion classes

Table 12 The *Precision* comparison results of ablation experiments on the WBEmoDialog dataset

Model	*Happi.*	*Love*	*Sorrow*	*Fear*	*Disgust*	*None*	MP	WP
w/o ET	0.812	0.624	0.416	0.293	0.632	0.720	0.583	0.684
w/o IDM	0.884	**0.653**	0.444	0.448	0.659	0.701	0.632	0.700
w/o ELMo	0.866	0.622	**0.496**	0.463	**0.717**	0.688	0.642	0.696
SINN	**0.892**	0.636	0.396	**0.600**	0.659	**0.733**	**0.650**	**0.711**

Happi. represents *Happiness*. MP means the macro average of *Precision* across all the emotion classes, and WP means the weighted average of *Precision* across all the emotion classes

target utterance. Thus w/o ET can also provide a better comparison for the models without previous emotion labels.

w/o IDM This is the SINN model without the Interactive Dependency Matching component.

w/o ELMo We replace the ELMo in SINN with randomly initialized word embeddings.

In Tables 10, 11, 12 and 13, we can observe obvious performance degradation in w/o ET and w/o IDM in terms of weighted average *Precision* and *F-Score* compared with the full model SINN. This indicates that either Empathy Tracking component or Interactive Dependency Matching component can provide important inter-speaker clues to enhance the representations of historical utterances. So the integrated model owns more ability than the separate parts and each part plays an indispensable role in the model's performance. On the other hand, the performance of average metrics also decrease when we replace ELMo with randomly initialized word embeddings. This validates the effectiveness of pre-trained language model such as ELMo for the text classification tasks.

We compare w/o ET with other baseline methods in Figure 9. It can be seen that in the DailyDialog dataset (i.e. the left figure) w/o ET outperforms baseline models in terms of average weighted *F-Score*, indicating that the proposed SINN is also effective without emotion labels. However, in the WBEmoDialog dataset (i.e. the right figure), w/o ET does not achieve better performance than the CMN and DialogGCN. The WBEmoDialog dataset is smaller compared with the DailyDialog dataset. We conjecture that fewer training instances limit the model's predicting performance because the proposed SINN has much more parameters. Although DialogGCN does not incorporate previous emotion labels, the

Table 13 The *F-Score* comparison results of ablation experiments on the WBEmoDialog dataset

Model	*Happi.*	*Love*	*Sorrow*	*Fear*	*Disgust*	*None*	MF	WF
w/o ET	0.809	0.594	0.388	0.321	0.537	0.762	0.569	0.684
w/o IDM	0.825	0.596	0.338	0.338	0.576	0.773	0.574	0.695
w/o ELMo	**0.841**	**0.608**	**0.422**	**0.427**	0.462	0.757	0.589	0.683
SINN	0.832	0.593	0.356	0.353	**0.643**	**0.779**	**0.593**	**0.708**

Happi. represents *Happiness*. MF means the macro average of *F-Score* across all the emotion classes, and WF means the weighted average of *F-Score* across all the emotion classes

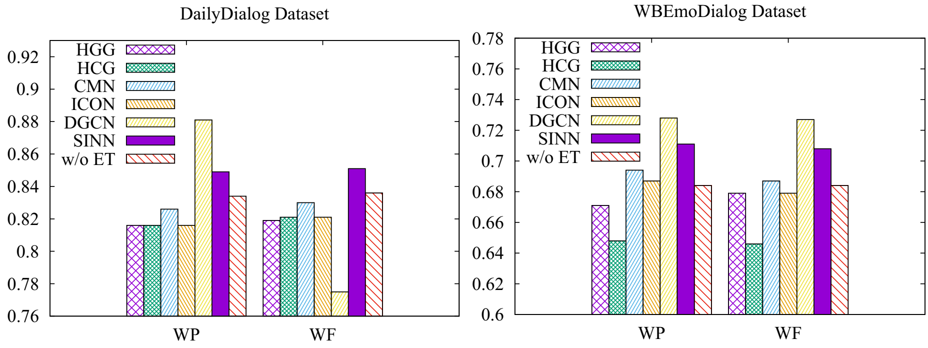


Figure 9 The performance comparisons of weighted *Precision* and *F-Score* for SINN, SINN w/o ET and other baseline methods. WA means the weighted average of *Precision* across all the emotion classes, and WF means the weighted average of *F-Score* across all the emotion classes. DGCN represents the Dialog-GCN model. Note that the baseline models (HGG, HCG, CMN, ICON, DGCN) do not include emotion labels of previous utterances. This figure demonstrates the effectiveness of the empathy tracking module in the SINN model, and also provides fair comparisons of SINN w/o ET with other baseline methods regardless of previous emotion labels

graph in DialogGCN is built based on not only backward but also forward context of the target utterances, which captures richer context information than the other models.

7 Conclusion

In this paper, we propose a novel SINN method that modeling the self and inter-speaker influences to identify the emotions in the conversations. Our proposed SINN can extract the deep inter-speaker influences from two effective components and merge them with the target utterance. Moreover, we adopt multiple attention mechanisms to help our model to pick up important information for predicting the final emotion. For better evaluating the proposed model, we construct a Chinese conversation dataset WBEmoDialog, which has more than 10 thousand conversations and 45 thousand utterances with emotion labels. Extensive experiments on the publicly available dataset DailyDialog and our constructed dataset WBEmoDialog. The results show that our proposed SINN model outperforms baseline methods with large margins on the DailyDialog dataset. When the label distribution is more balanced in the WBEmoDialog, the strongest baseline DialogGCN demonstrates the performance advantages over our SINN model. This work can also be extended to multi-participant conversations, which is left to our future work.

Acknowledgements The work was supported by the National Key R&D Program of China under Grant 2018YFB1004700, National Natural Science Foundation of China (61872074, 61772122), and the Fundamental Research Funds for the Central Universities (N180716010). This paper is a substantial extension of our previous work in [48]. In this paper, a large-scale Chinese dialog dataset WBEmoDialog is constructed, and the proposed SINN model is evaluated on the new dataset. More baselines, ablation experiments and discussions are included in the extended version. We thank reviewers for their valuable comments and suggestions.

References

1. Akbari, H., Sadiq, M.T., Rehman, A.U.: Classification of normal and depressed EEG signals based on centered correntropy of rhythms in empirical wavelet transform domain. Health Inf. Sci. Syst. **9**(1), 9 (2021)
2. Becker, K., Moreira, V.P., dos Santos, A.G.L.: Multilingual emotion classification using supervised learning: Comparative experiments. Inf. Process. Manag. **53**(3), 684–704 (2017)
3. Busso, C., Bulut, M., Lee, C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J.N., Lee, S., Narayanan, S.S.: IEMOCAP: interactive emotional dyadic motion capture database. Lang. Resour. Eval. **42**(4), 335–359 (2008)
4. Chatterjee, A., Gupta, U., Chinnakotla, M.K., Srikanth, R., Galley, M., Agrawal, P.: Understanding emotions in text using deep learning and big data. Comput. Hum. Behav. **93**, 309–317 (2019)
5. Cho, K., van Merrienboer, B., Gülċehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 1724–1734 (2014)
6. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv:1406.1078 (2014)
7. Ekman, P.: An argument for basic emotions. Cogn. Emot. **6**(3-4), 169–200 (1992)
8. Feng, J., Rao, Y., Xie, H., Wang, F.L., Li, Q.: User group based emotion detection and topic discovery over short text. World Wide Web **23**(3), 1553–1587 (2020)
9. Feng, S., Wang, Y., Liu, L., Wang, D., Yu, G.: Attention based hierarchical LSTM network for context-aware microblog sentiment classification. World Wide Web **22**(1), 59–81 (2019)
10. Feng, S., Wang, Y., Song, K., Wang, D., Yu, G.: Detecting multiple coexisting emotions in microblogs with convolutional neural networks. Cogn. Comput. **10**(1), 136–155 (2018)
11. Ferraro, G., Gee, B.L., Ji, S., Salvador-Carulla, L.: Lightme: analysing language in internet support groups for mental health. Health Inf. Sci. Syst. **8**(1), 34 (2020)
12. Fung, P., Bertero, D., Wan, Y., Dey, A., Chan, R.H.Y., Siddique, F.B., Yang, Y., Wu, C., Lin, R.: Towards empathetic human-robot interactions. In: Computational Linguistics and Intelligent Text Processing - 17th International Conference, CICLing 2016, Konya, Turkey, April 3-9, 2016, Revised Selected Papers, Part II, pp. 173–193 (2016)
13. Gui, L., Lin, H., Lin, Y., Liu, S.: Detection and extraction of hot topics on chinese microblogs. Cogn. Comput. **8**(4), 577–586 (2016)
14. Gupta, U., Chatterjee, A., Srikanth, R., Agrawal, P.: A sentiment-and-semantics-based approach for emotion detection in textual conversations. arXiv:1707.06996 (2017)
15. Hazarika, D., Poria, S., Mihalcea, R., Cambria, E., Zimmermann, R.: Icon: Interactive conversational memory network for multimodal emotion detection. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 2594–2604 (2018)
16. Hazarika, D., Poria, S., Zadeh, A., Cambria, E., Morency, L.P., Zimmermann, R.: Conversational memory network for emotion recognition in dyadic dialogue videos. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 2122–2132 (2018)
17. Hill, C.E., O'Brien, K.M.: Helping skills: Facilitating exploration, insight, and action. American Psychological Association, Washington (1999)
18. Hossain, M.D., Kabir, M.A., Anwar, A., Islam, M.Z.: Detecting autism spectrum disorder using machine learning techniques. Health Inf. Sci. Syst. **9**(1), 17 (2021)
19. Hsu, C., Chen, S., Kuo, C., Huang, T.K., Ku, L.: Emotionlines: An emotion corpus of multi-party conversations. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018 (2018)
20. Huang, M., Cao, Y., Dong, C. arXiv:1605.01478 (2016)
21. Husin, N., Abdullah, M.T., Mahmod, R.: A systematic literature review for topic detection in chat conversation for cyber-crime investigation. Int. J. Digit. Content Technol. Appl. **8**(3), 22 (2014)
22. Inui, K., Jiang, J., Ng, V., Wan, X. (eds.): Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019 Association for Computational Linguistics (2019)

23. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 1746–1751 (2014)
24. Kuppens, P., Allen, N.B., Sheeber, L.B.: Emotional inertia and psychological maladjustment. Psychol. Sci. **21**(7), 984–991 (2010)
25. Li, Y., Su, H., Shen, X., Li, W., Cao, Z., Niu, S.: Dailydialog: A manually labelled multi-turn dialogue dataset. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017 - Volume 1: Long Papers, pp. 986–995 (2017)
26. Liu, S., Lee, I.: Extracting features with medical sentiment lexicon and position encoding for drug reviews. Health Inf. Sci. Syst. **7**(1), 11 (2019)
27. Liu, S., Zheng, C., Demasi, O., Sabour, S., Li, Y., Yu, Z., Jiang, Y., Huang, M.: Towards emotional support dialog systems. arXiv:2106.01144 (2021)
28. Luo, L., Yang, H., Chin, F.Y.: Emotionx-dlc: Self-attentive bilstm for detecting sequential emotions in dialogue. arXiv:1806.07039 (2018)
29. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings (2013)
30. Morency, L., Bohus, D., Aghajan, H.K., Cassell, J., Nijholt, A., Epps, J. (eds.): International Conference on Multimodal Interaction, ICMI ’12, Santa Monica, CA, USA, October 22-26, 2012. ACM (2012)
31. Morris, M.W., Keltner, D.: How emotions work: The social functions of emotional expression in negotiations. Res. Organ. Behav. **22**, 1–50 (2000)
32. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. arXiv:1802.05365 (2018)
33. Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., Mihalcea, R.: MELD: A multimodal multi-party dataset for emotion recognition in conversations. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers, pp. 527–536 (2019)
34. Purpura, A., Masiero, C., Silvello, G., Susto, G.A.: Supervised lexicon extraction for emotion classification. In: Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, pp. 1071–1078 (2019)
35. Rao, Y., Lei, J., Wenyin, L., Li, Q., Chen, M.: Building emotional dictionary for sentiment analysis of online news. World Wide Web **17**(4), 723–742 (2014)
36. Rashkin, H., Smith, E.M., Li, M., Boureau, Y.: Towards empathetic open-domain conversation models: A new benchmark and dataset. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers, pp. 5370–5381 (2019)
37. Ren, Y., Zhang, Y., Zhang, M., Ji, D.: Context-sensitive twitter sentiment classification using neural network. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
38. Shen, C., Sun, C., Wang, J., Kang, Y., Li, S., Liu, X., Si, L., Zhang, M., Zhou, G.: Sentiment classification towards question-answering with hierarchical matching network. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 3654–3663 (2018)
39. Shen, L., Feng, Y.: CDL: curriculum dual learning for emotion-controllable response generation. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, pp. 556–566 (2020)
40. Song, K., Bing, L., Gao, W., Lin, J., Zhao, L., Wang, J., Sun, C., Liu, X., Zhang, Q.: Using customer service dialogues for satisfaction analysis with context-assisted multiple instance learning. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, pp. 198–207 (2019)
41. Song, K., Feng, S., Gao, W., Wang, D., Chen, L., Zhang, C.: Build emotion lexicon from microblogs by combining effects of seed words and emoticons in a heterogeneous graph. In: Proceedings of the 26th ACM Conference on Hypertext & Social Media, HT 2015, Guzelyurt, TRNC, Cyprus, September 1-4, 2015, pp. 283–292 (2015)
42. Sun, H., Lin, Z., Zheng, C., Liu, S., Huang, M.: Psyqa: A chinese dataset for generating long counseling text for mental health support. arXiv:2106.01702 (2021)
43. Tago, K., Takagi, K., Kasuya, S., Jin, Q.: Analyzing influence of emotional tweets on user relationships using naive bayes and dependency parsing. World Wide Web **22**(3), 1263–1278 (2019)
44. Thabtah, F.A., Abdelhamid, N., Peebles, D.: A machine learning autism classification based on logistic regression analysis. Health Inf. Sci. Syst. **7**(1), 12 (2019)

45. Tokhisa, R., Inui, K., Matsumoto, Y.: Emotion classification using massive examples extracted from the web. In: COLING 2008, 22nd International Conference on Computational Linguistics, Proceedings of the Conference, 18-22 August 2008, Manchester, UK, pp. 881–888 (2008)
46. Vanzo, A., Croce, D., Basili, R.: A context-based model for sentiment analysis in twitter. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pp. 2345–2354 (2014)
47. Wang, Y., Feng, S., Wang, D., Zhang, Y., Yu, G.: Context-aware chinese microblog sentiment classification with bidirectional lstm. In: Asia-Pacific Web Conference, pp. 594–606. Springer (2016)
48. Wei, J., Feng, S., Wang, D., Zhang, Y., Li, X.: Attentional neural network for emotion detection in conversations with speaker influence awareness. In: Tang, J., Kan, M., Zhao, D., Li, S., Zan, H. (eds.) Natural Language Processing and Chinese Computing - 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9-14, 2019, Proceedings, Part II, Lecture Notes in Computer Science, vol. 11839, pp. 287–297. Springer (2019)
49. Wen, S., Wan, X.: Emotion classification in microblog texts using class sequential rules. In: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, pp. 187–193 (2014)
50. Yang, Y., Zhou, D., He, Y., Zhang, M.: Interpretable relevant emotion ranking with event-driven attention. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, pp. 177–187 (2019)
51. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480–1489 (2016)
52. Zhang, L., Chen, C.: Sentiment classification with convolutional neural networks: An experimental study on a large-scale chinese conversation corpus. In: 2016 12th International Conference on Computational Intelligence and Security (CIS), pp. 165–169. IEEE (2016)
53. Zhang, Y., Fu, J., She, D., Zhang, Y., Wang, S., Yang, J.: Text emotion distribution learning via multi-task convolutional neural network. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden, pp. 4595–4601 (2018)
54. Zhou, H., Huang, M., Zhang, T., Zhu, X., Liu, B.: Emotional chatting machine: Emotional conversation generation with internal and external memory. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pp. 730–739 (2018)
55. Zhou, X., Wang, W.Y.: Mojitalk: Generating emotional responses at scale. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers, pp. 1128–1137 (2018)
56. Zhou, Y., Li, C., Xu, B., Xu, J., Yang, L., Xu, B.: Constructing a Chinese conversation corpus for sentiment analysis. In: Natural Language Processing and Chinese Computing - 6th CCF International Conference, NLPCC 2017, Dalian, China, November 8-12, 2017, Proceedings, pp. 579–590 (2017)

Affiliations

Shi Feng[1] · **Jia Wei**[1] · **Daling Wang**[1] · **Xiaocui Yang**[1] · **Zhenfei Yang**[1] · **Yifei Zhang**[1] · **Ge Yu**[1]

Jia Wei
weijia_neu@163.com

Daling Wang
wangdaling@cse.neu.edu.cn

Xiaocui Yang
yangxiaocui@stumail.neu.edu.cn

Zhenfei Yang
2001851@stu.neu.edu.cn

Yifei Zhang
zhangyifei@cse.neu.edu.cn

Ge Yu
yuge@cse.neu.edu.cn

[1] School of Computer Science and Engineering, Northeastern University,
No.195 Chuangxin Road, Hunnan District, Shenyang, China