An Iterative Emotion Interaction Network for Emotion Recognition in Conversations

Xin Lu, Yanyan Zhao, Yang Wu, Yijian Tian, Huipeng Chen, Bing Qin*
Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, China
{xlu, yyzhao, ywu, yjtian, chp, qinb}@ir.hit.edu.cn

Abstract

Emotion recognition in conversations (ERC) has received much attention recently in the natural language processing community. Considering that the emotions of the utterances in conversations are interactive, previous works usually implicitly model the emotion interaction between utterances by modeling dialogue context, but the misleading emotion information from context often interferes with the emotion interaction. We noticed that the gold emotion labels of the context utterances can provide explicit and accurate emotion interaction, but it is impossible to input gold labels at inference time. To address this problem, we propose an iterative emotion interaction network, which uses iteratively predicted emotion labels instead of gold emotion labels to explicitly model the emotion interaction. This approach solves the above problem, and can effectively retain the performance advantages of explicit modeling. We conduct experiments on two datasets, and our approach achieves state-of-the-art performance.

1 Introduction

Emotion recognition in conversations (ERC) aims to recognize the emotion of each utterance in conversations. Recently it has received much attention due to its applications in various conversation scenes, such as emotional chatbots (Zhou et al., 2018), emotion detection of customers in artificial services (Song et al., 2019), sentiment analysis of comments in social media (Chatterjee et al., 2019), and so on.

Different from the common sentence-level emotion recognition task, ERC is special due to some characteristics. The first one is that the utterances are context dependent, and modeling context can provide more information for emotion recognition (Poria et al., 2017; Jiao et al., 2019). The second characteristic of ERC is that the utterances are speaker-sensitive, thus many researchers modeled the state of speakers and the inter-speaker dependency relations (Hazarika et al., 2018b; Majumder et al., 2019; Zhang et al., 2019; Ghosal et al., 2019). In this paper, we observe another characteristic that is the emotions of the utterances are interactive. For example, in Figure 1, the emotion of the utterance from Speaker A can directly influence Speaker B. Thus, modeling the emotion interaction between utterances is helpful for the ERC task.

Previous works usually implicitly model the emotion interaction by modeling dialogue context (Poria et al., 2017; Jiao et al., 2019). However, because of the arbitrariness of the dialogue, the context utterances often convey misleading emotion information when recognizing the emotion of the target utterance, such as in Figure 1(a). To solve this problem, we observe that the gold emotion labels (such as "happy", "angry") of the context utterances can provide explicit and accurate emotion interaction between utterances, such as in Figure 1(b). Thus, we can introduce the emotion labels to explicitly model the emotion interaction between utterances.

However, a challenging problem is this approach requires inputting gold labels of context utterances, which is impossible at inference time. We observe a phenomenon, the utterances which are helpful for the emotion recognition of the target utterance are usually near the target utterance and the number is

* Email corresponding.

Proceedings of the 28th International Conference on Computational Linguistics, pages 4078–4088
Barcelona, Spain (Online), December 8-13, 2020

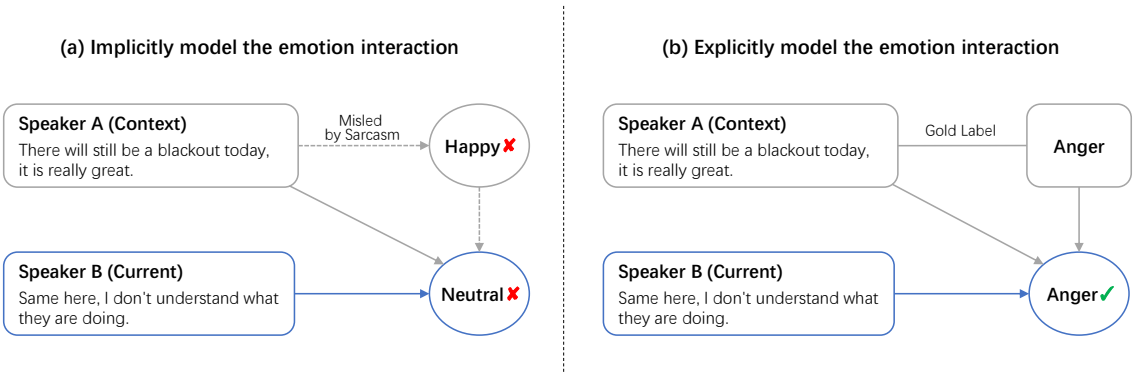

Figure 1: A short conversation example which shows the difference between two methods of modeling emotion interaction. (a) The emotion prediction of Speaker B is wrong, because of the interference from Speaker A's sarcasm. (b) Due to modeling emotion interaction explicitly, the interference from Speaker A's sarcasm is reduced and the emotion prediction of Speaker B is right.

often limited. As long as the emotion labels of these utterances are correct, the emotion recognition of the target utterance can benefit from our explicit approach. Therefore, we speculate that even if only part of emotion labels are correct, the correct part can still help related utterances recognize their emotions better. This process can be iterated, which will make the emotion recognition result better and better. Experiments in Section 3.5 also confirm this.

Based on the above idea, we propose an iterative emotion interaction network for emotion recognition in conversations. This network explicitly models the emotion interaction between utterances, and meanwhile solves the problem of no gold labels at inference time by iterative improvement mechanism. Specifically, we first adopt an utterance encoder to obtain the representations of utterances and make an initial prediction for the emotions of all utterances. Next, we integrate the initial prediction and the utterances by an emotion interaction based context encoder to make an updated prediction for the emotions. Finally, we use the iterative improvement mechanism to iteratively update the emotions, in which a loss function is employed to constrain the prediction of each iteration and the correction behavior between two adjacent iterations.

The contributions of this work are summarized as follows:

- We explicitly model the emotion interaction between utterances, which is superior to the previous works implicitly modeling the emotion interaction.
- We propose an iterative emotion interaction network, which not only explicitly models the emotion interaction between utterances, but also solves the problem of no gold labels at inference time.
- We conduct experiments on the IEMOCAP dataset and the MELD dataset. Experimental results show that our approach achieves state-of-the-art performance.

2 Method

In this section, we introduce our proposed iterative emotion interaction network as shown in Figure 2. Our network consists of three components: an utterance encoder, an emotion interaction based context encoder, and iterative improvement mechanism. The utterance encoder is used to obtain the representations of all utterances in a conversation. The emotion interaction based context encoder introduces the emotion probabilities of the utterances and integrates them and the utterance representations to explicitly model the emotion interaction. The iterative improvement mechanism contains initial emotion prediction, iterative emotion feedback and loss for iteration, which combines the above two encoders to iteratively improve the emotion predictions. In the following sections, we describe these components in detail.

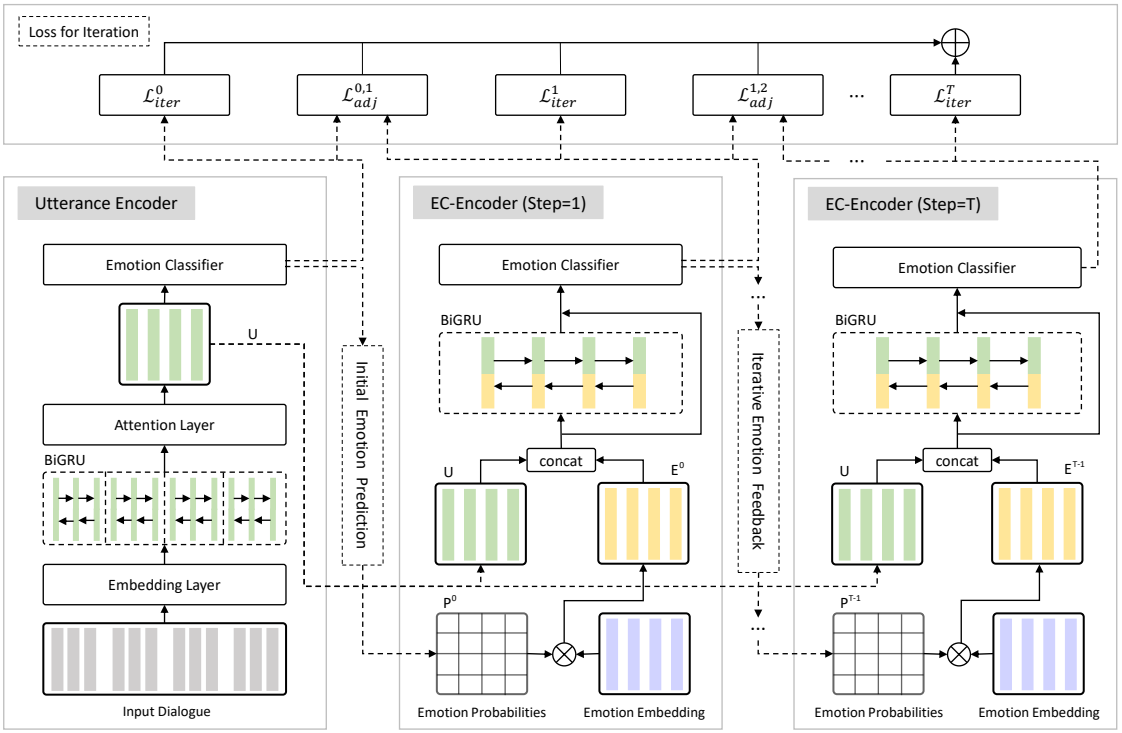

Figure 2: Overview of our proposed iterative emotion interaction network. The utterance encoder is used to obtain the utterance representations. The emotion interaction based context encoder (EC-Encoder) is used to explicitly model the emotion interaction and obtain the updated emotion probabilities. The iterative improvement mechanism (including the initial emotion prediction, the iterative emotion feedback and the loss for iteration) is used to build the iterative framework and calculate the loss for iteration. These components work together and finally improve the performance.

2.1 Utterance Encoder

In our framework, the goal of the utterance encoder is to obtain the representation for each utterance. Suppose, given an utterance $u = \{w_1, w_2, ..., w_M\}$ consisting of a sequence of M words, we first obtain the embedding forms $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M\}$ by fedding them into the word embedding layer, which is initialized by pretrained word embeddings. A BiGRU is used to capture the contextual information from $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M\}$, yielding two sequences of hidden states $\{\overrightarrow{\mathbf{h}_1}, \overrightarrow{\mathbf{h}_2}..., \overrightarrow{\mathbf{h}_M}\}$ and $\{\overleftarrow{\mathbf{h}_1}, \overleftarrow{\mathbf{h}_2}, ..., \overleftarrow{\mathbf{h}_M}\}$. We concatenate $\overrightarrow{\mathbf{h}_i}$ and $\overleftarrow{\mathbf{h}_i}$ into a single vector $\mathbf{h}_i$ for the word w_i, which is defined as follows:

$$\mathbf{h}_i = [\overrightarrow{\mathbf{h}_i}; \overleftarrow{\mathbf{h}_i}] \tag{1}$$

To obtain a single vector representation for the utterance u, we aggregate the sequence of hidden states $\{\mathbf{h}_1, \mathbf{h}_2..., \mathbf{h}_M\}$ with an attention mechanism, which can be formulated as follows:

$$\alpha_i = \frac{\exp(\mathbf{h}_i^\top \mathbf{W}_u)}{\sum_j \exp(\mathbf{h}_j^\top \mathbf{W}_u)} \tag{2}$$

$$\mathbf{u} = \sum_{i=1}^{M} \alpha_i \mathbf{h}_i \tag{3}$$

where $\mathbf{u}$ is the vector representation for the utterance u. Similarly, given a conversation $C = \{u_1, u_2, ..., u_N\}$, the sequence of all utterances can be represented as $\mathbf{U} = \{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_N\}$.

2.2 Emotion Interaction Based Context Encoder

The emotion interaction based context encoder is used to explicitly model the emotion interaction. It introduces the emotion probabilities of the utterances, and integrates them and the utterance representations to achieve this goal. It consists of three components: an emotion embedding layer, a BiGRU encoder, and an emotion classifier. It takes the utterance representations $\mathbf{U} = \{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_N\}$ and the context emotion probabilities $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_N\}$ as inputs and then outputs the updated version of $\mathbf{P}$, named $\mathbf{P}^{'}$. Thus, it is also the basic unit of iterative improvement in our framework.

Emotion Embedding Let $L = \{l_1, l_2, ..., l_{|L|}\}$ represents the set of emotion labels, and then map each label l_i to an embedding vector $\mathbf{x}_i$ which is the representation of this emotion. For each utterance emotion probability vector $\mathbf{p}_i \in \mathbf{P}$, we define $\mathbf{p}_i = \{p_i^1, p_i^2, ..., p_i^{|L|}\}$ and then use these as weights to obtain the utterance emotion representation $\mathbf{e}_i$, which is a weighted sum of all emotion embeddings:

$$\mathbf{e}_i = \sum_{j=1}^{|L|} p_i^j \mathbf{x}_j \tag{4}$$

Based on the above, we can obtain the context emotion representations $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_N\}$.

BiGRU Encoder For each utterance u_i, we concatenate $\mathbf{u}_i \in \mathbf{U}$ and $\mathbf{e}_i \in \mathbf{E}$, and feed the result into GRU units. The process can be defined as follows:

$$\overrightarrow{\mathbf{h}_i} = \overrightarrow{\mathbf{GRU}}([\mathbf{u}_i; \mathbf{e}_i], \overrightarrow{\mathbf{h}_{i-1}}) \tag{5}$$

$$\overleftarrow{\mathbf{h}_i} = \overleftarrow{\mathbf{GRU}}([\mathbf{u}_i; \mathbf{e}_i], \overleftarrow{\mathbf{h}_{i+1}}) \tag{6}$$

$$\mathbf{h}_i = \overrightarrow{\mathbf{h}_i} + \overleftarrow{\mathbf{h}_i} + [\mathbf{u}_i; \mathbf{e}_i] \tag{7}$$

where $\mathbf{h}_i$ is the hidden state, which is included in the context hidden states $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_N\}$.

Emotion Classifier For each $\mathbf{h}_i \in \mathbf{H}$, we feed it into the emotion classifier which is a softmax layer:

$$\mathbf{p}_i^{'} = \mathrm{softmax}(\mathbf{W}_e \mathbf{h}_i + \mathbf{b}_e) \tag{8}$$

where $\mathbf{p}_i^{'}$ is the updated emotion probability vector, which is included in the updated context emotion probabilities $\mathbf{P}^{'}$.

2.3 Iterative Improvement Mechanism

The iterative improvement mechanism is the core of our proposed approach. It consists of three parts: initial emotion prediction, iterative emotion feedback, and loss for iteration. These three parts combines the above two encoders to build an iterative framework, which can iteratively improve the emotion predictions. In this section, we introduce these three parts in detail.

Initial Emotion Prediction Generally, the initial value is an important part of the iteration. In our framework, we obtain the initial context emotion probabilities $\mathbf{P}^0$ by feeding the utterance representations $\mathbf{U}$ into a softmax layer. The process can be defined as follows:

$$\mathbf{p}_i^0 = \mathrm{softmax}(\mathbf{W}_p \mathbf{u}_i + \mathbf{b}_p) \tag{9}$$

where $\mathbf{u}_i$ is an utterance representation from $\mathbf{U}$, and $\mathbf{p}_i^0$ is the initial emotion probability vector which should be contained in $\mathbf{P}^0$.

Iterative Emotion Feedback This component is crucial to achieve iterative improvement. As mentioned in Section 2.2, the basic iterative unit is the emotion interaction based context encoder (EC-Encoder), which takes the context emotion probabilities as input and outputs an updated version. The iterative emotion feedback mainly uses the updated context emotion probabilities as the input of the EC-Encoder again, thereby achieves an iterative update of the emotion prediction.

Formally, the process of obtaining the updated context emotion probabilities in the i-th step can be defined as follows:

$$\mathbf{P}^i = \text{EC-Encoder}(\mathbf{P}^{i-1}, \mathbf{U}) \tag{10}$$

where $i \geq 1$, $\mathbf{U}$ is the utterance representations, $\mathbf{P}^{i-1}$ is the context emotion probabilities at step i-1, and $\mathbf{P}^i$ is the context emotion probabilities at step i.

Loss for Iteration To achieve iterative improvement, we design a loss to constrain the prediction of each iteration and the correction behavior between two adjacent iterations.

For each iteration, we use cross-entropy function to obtain the loss:

$$\mathcal{L}^i_{iter} = -\frac{1}{N_a}\sum_{j=1}^{N_a}\sum_{k=1}^{|L|} y_{j,k}\log(p^i_{j,k}) \tag{11}$$

We add margin-ranking loss between two adjacent iterations, which can punish incorrect modification:

$$\mathcal{L}^{i,\ i+1}_{adj} = \frac{1}{N_a}\sum_{j=1}^{N_a}\sum_{k=1}^{|L|} y_{j,k}\max(0, p^i_{j,k} - p^{i+1}_{j,k}) \tag{12}$$

The final loss can be defined as follows:

$$\mathcal{L} = \frac{1}{T+1}\sum_{i=0}^{T}\mathcal{L}^i_{iter} + \lambda * \frac{1}{T}\sum_{i=0}^{T-1}\mathcal{L}^{i,\ i+1}_{adj} \tag{13}$$

where T is a hyperparameter which represents the number of iterations, N_a is the number of all utterances in the dataset, and $|L|$ is the number of emotions labels. $\mathbf{y}_j$ denotes the one-hot vector of gold labels, and $y_{j,k}$ is the element of $\mathbf{y}_j$ for emotion k. Similarly, $p^i_{j,k}$ and $p^{i-1}_{j,k}$ are the elements of $\mathbf{p}^i_j$ and $\mathbf{p}^{i-1}_j$ for emotion k. In addition, λ is a hyperparameter that balances two types of losses.

3 Experiments

3.1 Datasets

We evaluate the performance of our approach on two publicly available datasets, IEMOCAP (Busso et al., 2008) and MELD (Poria et al., 2019).

IEMOCAP The IEMOCAP dataset[1] was collected by SAIL lab at USC. It consists of approximately 12 hours of multimodal conversation data, we only use the text modality in this paper. It is grouped into five sessions, we use the first four sessions as the training set and use the last one as the test set. Besides, we use 10% dialogues of the training set as a validation set. The dataset contains 152 dialogues with a total of 7,433 utterances, and it comes with six emotion categories.

MELD The MELD dataset[2] contains the conversations from Friends TV show transcripts, which is a multimodal extension of the EmotionLines dataset (Hsu et al., 2018). In this paper, we only use the text modality. The dataset contains 1,433 dialogues with a total of 13,708 utterances, and it comes with seven emotion categories.

3.2 Baselines

We compare our approach with the following baselines:

CNN (Kim, 2014) This is a CNN model trained on context-independent utterances, hence the contextual information is unused in this baseline.

cLSTM (Poria et al., 2017) This is a context-level LSTM model. This baseline uses CNN to extract context-independent utterance features, and uses LSTM to capture contextual features for emotion recognition.

[1]https://sail.usc.edu/iemocap/

[2]https://affective-meld.github.io/

Models	IEMOCAP						
	happy	*sad*	*neutral*	*anger*	*excited*	*frustrated*	***w-Avg.***
CNN	32.91	50.41	52.33	55.24	46.84	54.51	50.15
cLSTM	30.66	69.86	55.15	58.52	55.93	60.74	57.01
cLSTM+CRF	35.71	69.59	56.43	62.44	50.34	60.23	56.98
DialogueRNN	38.74	76.08	58.26	**63.10**	68.75	60.37	62.15
DialogueGCN	51.87	76.76	56.76	62.26	**72.71**	58.04	63.16
Our Approach	**53.17**	**77.19**	**61.31**	61.45	69.23	**60.92**	**64.37**

Table 1: Performance of different approaches on the IEMOCAP dataset.

Models	MELD							
	neutral	*surprise*	*fear*	*sadness*	*joy*	*disgust*	*anger*	***w-Avg.***
CNN	77.24	50.54	0.32	22.28	54.19	2.86	42.88	58.48
cLSTM	76.47	50.17	0.92	**26.51**	55.62	9.65	46.77	59.33
cLSTM+CRF	76.42	50.22	1.48	26.29	55.58	8.51	46.96	59.29
DialogueRNN	76.23	49.59	0.00	26.33	54.55	0.81	46.76	58.73
DialogueGCN	76.02	46.37	0.98	24.32	53.62	1.22	43.03	57.52
Our Approach	**77.52**	**53.65**	**3.31**	23.62	**56.63**	**19.38**	**48.88**	**60.72**

Table 2: Performance of different approaches on the MELD dataset.

cLSTM+CRF This is a modified model based on cLSTM (Poria et al., 2017). We add a CRF (Conditional Random Fields) layer after the contextual LSTM, so that this baseline could capture the dependencies between emotion labels.

DialogueRNN (Majumder et al., 2019) This is a RNN-based model, which uses three GRUs to model the speaker, the context given by the preceding utterances, and the emotion behind the preceding utterances. This baseline could set separate states for each speaker and associate states with the speaker's utterance. In our experiment, we use the open-source codes[3] of DialogueRNN provided by the authors.

DialogueGCN (Ghosal et al., 2019) This is a GCN-based model. This baseline uses a GCN to model the conversation, the nodes in the graph represent utterances, and the types of edges are determined based on the speaker information. In our experiment, we use the open-source codes[4] of DialogueGCN provided by the authors.

3.3 Experimental Settings

In our experiment setting, we use the pretrained 840B GloVe embedding (Pennington et al., 2014) to initialize the 300 dimensional word embedding layer, and we set the emotion embedding dimension to 32. In utterance encoder, the hidden size of GRU is 50 for IEMOCAP and 100 for MELD. In emotion interaction based context encoder, the hidden size of GRU for two datasets is 132 and 232 respectively.

We use Adam (Kingma and Ba, 2015) to optimize the parameters in our models, and use a mini-batch size of 32. To regulate our models, we set the weight decay to 0.0001, and apply dropout with a dropout rate at 0.1. Based on validation performance on IEMOCAP, the learning rate is set to 0.0002, the hyperparameter λ is set to 50, and the maximum iteration number T is set to 3. Based on validation performance on MELD, the learning rate is set to 0.0001, the hyperparameter λ is set to 5, and the maximum iteration number T is set to 2.

[3]https://github.com/SenticNet/conv-emotion/tree/master/DialogueRNN/
[4]https://github.com/SenticNet/conv-emotion/tree/master/DialogueGCN/

3.4 Overall Results

We compare our approach with the baseline methods on IEMOCAP and MELD datasets, and the experimental results are shown in Table 1 and Table 2 respectively. We report the F1-score for each emotion class, and evaluate the overall performance using weighted average F1. For each result of our approach, we repeat the experiment 12 times to get the average value. For fair comparison, we re-run all baseline methods with the same setting. Therefore, the results of baseline methods are slightly different from those in original papers.

Table 1 presents the results on IEMOCAP dataset. Among all baseline methods, DialogueGCN achieves the best overall performance of 63.16% on weighted F1 score. In comparison, the performance of our approach outperforms DialogueGCN by 1.21%, which can preliminarily prove the effectiveness of our proposed approach. In addition, our approach obtains improvements on most emotion classes compared to baseline methods, although some performance degradations occur on *anger* and *excited*. But overall, our approach balances them well and improves overall performance.

Table 2 presents the results on MELD dataset. The cLSTM model achieves the best overall performance of 59.33% on weighted F1 score among all baseline methods. In comparison, the performance of our approach outperforms cLSTM by 1.39%. Similar to the results on IEMOCAP dataset, our approach also achieves the best performance on most emotion classes. In particular, though emotion class *disgust* only contains a few utterances, our approach improved the performance greatly (about 10%), which shows that our approach has the capability to recognize the emotions of minority classes.

3.5 Analysis

Our proposed approach achieves state-of-the-art performance. In this section, we analyze our approach from the following aspects.

Effectiveness of Emotion Interaction We analyze the effectiveness of modeling the emotion interaction on both datasets, and the experimental results are shown in the Table 3. First, we train a model based on our proposed network without emotion embedding and iterative emotion feedback, denoted *No Label*. This model represents an extreme case where the emotion labels are not used at all, which is the case of most implicit modeling emotion interaction methods. Second, we train a model without iterative emotion feedback but initialize context emotion representations with gold emotion labels, denoted *Gold Label*. This model represents another extreme case where the emotion labels are optimally used, which is the best way to explicitly model the emotion interaction, but it is impossible at inference time. From the results, we can see that: 1) The performance of *No Label* is the worst, the performance of *Gold Label* is the best, indicating that explicit modeling has more advantages than implicit modeling. 2) The performance of our approach falls somewhere in between, indicating that our iterative improvement mechanism can not only solve the problem of no gold labels, but also effectively retain the performance advantages of explicit modeling.

Models	**IEMOCAP**	**MELD**
No Label	60.22	59.91
Our Approach, $iter$ = 1	61.22	60.07
Our Approach, $iter$ = 2	63.66	**60.72**
Our Approach, $iter$ = 3	**64.37**	60.64
Gold Label	66.75	62.28

Table 3: An analysis of the effectiveness of emotion interaction on two datasets. Best values among our models are highlighted in bold.

Impact of Maximum Iteration Number We plot the performance trends of our approach with increasing the maximum iteration number on both datasets. As presented in Figure 3, the performance shows a trend of increases at first and then decreases, and the best performance is obtained when the

maximum iteration number is 3 for IEMOCAP and 2 for MELD. This result shows that appropriately increasing the maximum iteration number can gradually improve the performance, which is consistent with our expectation. However, too many iterations lead to a decrease in performance. This phenomenon is also consistent with our expectation, and one possible explanation is that too many iterations will lead to overfitting on the training set.

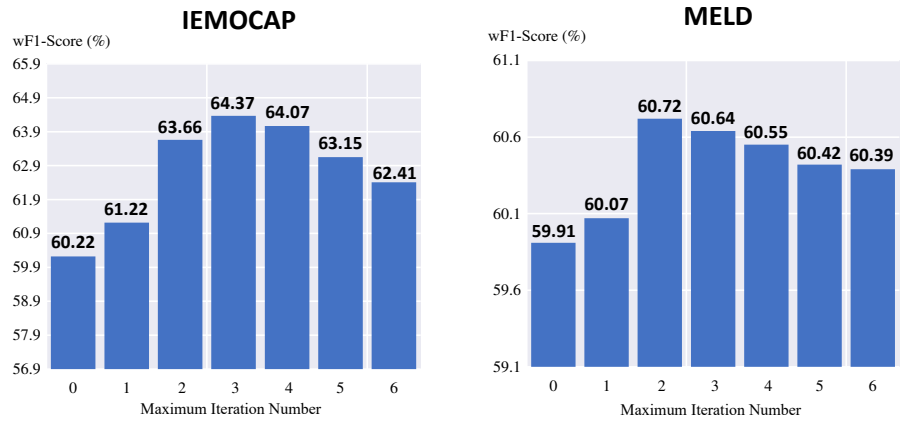

Figure 3: Performance trends with different maximum iteration number on two datasets.

Analysis of Iterative Correction Behavior We analyze the iterative correction behavior of our approach when the maximum iteration number is fixed, the performance of each step and the correction behavior between two adjacent steps are shown in the Table 4. For IEMOCAP and MELD datasets, we select the models with the maximum iteration number of 3 and 2 for analysis, respectively. From the results, we can see that: 1) The performance of each step gradually increases on both datasets, which shows that the iterative improvement mechanism works. 2) Among the changes of predicted emotion labels between all two adjacent steps on both datasets, the cases which are changed from wrong to right (**W** $\rightarrow$ **R**) are the most. This shows that our approach does make effective emotion prediction correction in the iterative process.

Dataset (Iteration)	Step	w-F1	Step $i \rightarrow$ Step j	R $\rightarrow$ W	W $\rightarrow$ R	W $\rightarrow$ W
IEMOCAP ($iter$=3)	$step$ = 1	61.97	$step$ 1 $\rightarrow$ $step$ 2	27.84%	**46.25%**	25.91%
	$step$ = 2	63.71				
	$step$ = 3	**64.37**	$step$ 2 $\rightarrow$ $step$ 3	27.81%	**47.78%**	24.41%
MELD ($iter$=2)	$step$ = 1	60.45	$step$ 1 $\rightarrow$ $step$ 2	32.53%	**39.86%**	27.61%
	$step$ = 2	**60.72**				

Table 4: An analysis of the iterative correction behavior on two datasets. **R** $\rightarrow$ **W** means the percentage of cases which are changed from right to wrong, **W** $\rightarrow$ **R** and **W** $\rightarrow$ **W** have similar meanings.

3.6 Case Study

We give a case study to illustrate the effectiveness of our proposed iterative improvement mechanism, as is shown in Table 5. We present a sample dialogue from the MELD test set and show the emotion labels predicted by our approach at each step, where the maximum iteration number is set to 2. It can be seen that the emotion prediction result of the first step has more errors, which is a less accurate result. As the iteration number increases, the situation of prediction errors is gradually corrected. Such as in the 8th utterance said by *Joey*, the emotion of this utterance is difficult to judge only based on its text, and the

No.	Speaker	Utterance	Step=1	Step=2	Gold
1	Chandler	What are you doing tonight?	neutral	neutral	neutral
2	Joey	Huh? Uh.	neutral	neutral	neutral
3	Chandler	Dude. Dude.	neutral	neutral	surprise
4	Joey	Oh, Sorry. Uh, I've got those plans with Phoebe, why?	neutral	neutral	neutral
5	Chandler	Oh really? Uh, Monica said she had a date at 9:00.	surprise	surprise	surprise
6	Joey	What? Tonight?	surprise	surprise	surprise
7	Chandler	That's what Monica said.	neutral	neutral	neutral
8	Joey	After she gave me that big speech?	neutral	surprise	surprise
9	Joey	She goes and makes a date on the same night she has plans with me?	neutral	anger	anger
10	Joey	I think she's trying to pull a fast one on Big Daddy.	anger	anger	anger

Table 5: An example of emotion prediction from the MELD test set output by our approach.

prediction of the first step is wrong. However, the prediction of the second step is modified to be correct, which is due to the context utterances and the *anger* emotion of the 10th utterance predicted correctly in the first step. This case shows that the iterative improvement mechanism is effective.

4 Related Work

Our work focuses on emotion recognition in conversations (ERC), which requires considering some characteristics in conversations. Early works on ERC noticed that dialogue context can provide more information. Poria et al. (2017) proposed the c-LSTM model, which used LSTM model to capture contextual features. Jiao et al. (2019) suggested the HiGRU model, which introduced a word-level GRU and an utterance-level GRU with self-attention and features fusion. Especially, Zhong et al. (2019) proposed the KET model, which introduced external commonsense knowledge to the ERC task. Qin et al. (2020) proposed the DCR-Net model, which improved the performance of the ERC task through multi-task learning. Recent works found that the state of the speakers and the inter-speaker dependency relations also need to be considered. These works can be divided into two categories: RNN-based models and GCN-based models. RNN-based models include CMN (Hazarika et al., 2018b), ICON (Hazarika et al., 2018a), and DialogueRNN (Majumder et al., 2019). CMN and ICON used different GRUs for both parties in the conversation and used memory networks to fuse the contextual information. DialogueRNN set separate states for each speaker and associated states with the speaker's utterance. GCN-based models include ConGCN (Zhang et al., 2019) and DialogueGCN (Ghosal et al., 2019). ConGCN represented each utterance and each speaker as a node and linked the utterances to the speakers by undirected edges. DialogueGCN also used a GCN to model the conversation. The graph constructed by DialogueGCN contains only utterance nodes, but the type of edge is determined based on the speaker information.

Different from these works, we focus on another characteristic that is the emotion interaction between utterances and propose an iterative emotion interaction network to explicitly model it. The related works usually model dialogue context, which only implicitly model the emotion interaction. Therefore, the motivations and practices of our work are different from the related works.

5 Conclusion

In this paper, we explicitly model the emotion interaction between utterances in ERC. To solve the problem of no gold emotion labels at inference time, we propose an iterative emotion interaction network, which uses iteratively predicted emotion labels instead of the gold emotion labels. The network consists of three components: the utterance encoder, the emotion interaction based context encoder, and the iterative improvement mechanism. These components work together and finally iteratively improve the emotion predictions. Experimental results on two datasets show that our approach achieves state-of-the-art performance, and extensive analysis further proves the effectiveness of our approach.

Acknowledgements

We thank the anonymous reviewers for their insightful comments and suggestions. This work was supported by the National Key R&D Program of China via grant 2018YFB1005103 and National Natural Science Foundation of China (NSFC) via grant 61632011 and 61772153.

References

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. 2008. IEMOCAP: interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42(4):335–359.

Ankush Chatterjee, Umang Gupta, Manoj Kumar Chinnakotla, Radhakrishnan Srikanth, Michel Galley, and Puneet Agrawal. 2019. Understanding emotions in text using deep learning and big data. *Computers in Human Behavior*, 93:309 – 317.

Deepanway Ghosal, Navonil Majumder, Soujanya Poria, Niyati Chhaya, and Alexander Gelbukh. 2019. DialogueGCN: A graph convolutional neural network for emotion recognition in conversation. 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)*, pages 154–164, Hong Kong, China, November. Association for Computational Linguistics.

Devamanyu Hazarika, Soujanya Poria, Rada Mihalcea, Erik Cambria, and Roger Zimmermann. 2018a. ICON: Interactive conversational memory network for multimodal emotion detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2594–2604, Brussels, Belgium, October-November. Association for Computational Linguistics.

Devamanyu Hazarika, Soujanya Poria, Amir Zadeh, Erik Cambria, Louis-Philippe Morency, and Roger Zimmermann. 2018b. 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)*, pages 2122–2132, New Orleans, Louisiana, June. Association for Computational Linguistics.

Chao-Chun Hsu, Sheng-Yeh Chen, Chuan-Chun Kuo, Ting-Hao K. Huang, and Lun-Wei Ku. 2018. Emotionlines: An emotion corpus of multi-party conversations. In Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Kôiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA).

Wenxiang Jiao, Haiqin Yang, Irwin King, and Michael R. Lyu. 2019. HiGRU: Hierarchical gated recurrent units for utterance-level emotion recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 397–406, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander F. Gelbukh, and Erik Cambria. 2019. Dialoguernn: An attentive RNN for emotion detection in conversations. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6818–6825. AAAI Press.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, and Louis-Philippe Morency. 2017. Context-dependent sentiment analysis in user-generated videos. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 873–883, Vancouver, Canada, July. Association for Computational Linguistics.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. 2019. MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 527–536, Florence, Italy, July. Association for Computational Linguistics.

Libo Qin, Wanxiang Che, Yangming Li, Minheng Ni, and Ting Liu. 2020. Dcr-net: A deep co-interactive relation network for joint dialog act recognition and sentiment classification. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8665–8672. AAAI Press.

Kaisong Song, Lidong Bing, Wei Gao, Jun Lin, Lujun Zhao, Jiancheng Wang, Changlong Sun, Xiaozhong Liu, and Qiong Zhang. 2019. 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)*, pages 198–207, Hong Kong, China, November. Association for Computational Linguistics.

Dong Zhang, Liangqing Wu, Changlong Sun, Shoushan Li, Qiaoming Zhu, and Guodong Zhou. 2019. Modeling both context- and speaker-sensitive dependence for emotion detection in multi-speaker conversations. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5415–5421. International Joint Conferences on Artificial Intelligence Organization, 7.

Peixiang Zhong, Di Wang, and Chunyan Miao. 2019. Knowledge-enriched transformer for emotion detection in textual conversations. 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)*, pages 165–176, Hong Kong, China, November. Association for Computational Linguistics.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *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*, pages 730–739. AAAI Press.