S+PAGE: A Speaker and Position-Aware Graph Neural Network Model for Emotion Recognition in Conversation

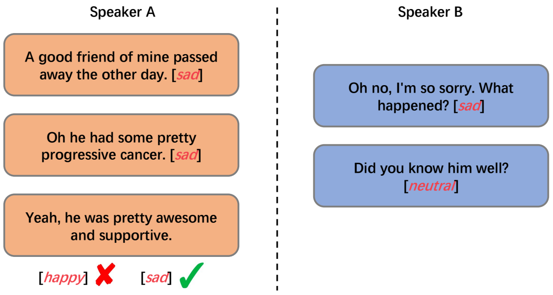
Chen Liang, Jing Xu, Yangkun Lin, Chong Yang[*], Yongliang Wang
Ant Group
Hangzhou, China
{liangchen.liangche, jill.xj, linyangkun.lyk,
yangchong.yang, yongliang.wyl}@antgroup.com

Abstract

Emotion recognition in conversation (ERC) has attracted much attention in recent years for its necessity in widespread applications. With the development of graph neural network (GNN), recent state-of-the-art ERC models mostly use GNN to embed the intrinsic structure information of a conversation into the utterance features. In this paper, we propose a novel GNN-based model for ERC, namely S+PAGE, to better capture the speaker and position-aware conversation structure information. Specifically, we add the relative positional encoding and speaker dependency encoding in the representations of edge weights and edge types respectively to acquire a more reasonable aggregation algorithm for ERC. Besides, a two-stream conversational Transformer is presented to extract both the self and inter-speaker contextual features for each utterance. Extensive experiments are conducted on four ERC benchmarks with state-of-the-art models employed as baselines for comparison, whose results demonstrate the superiority of our model.

1 Introduction

Emotion recognition in conversation (ERC), which aims to identify the emotion of each utterance in a conversation, is a task arousing increasing interests in many fields. With the prevalence of social media and intelligent assistants, ERC has great potential applications in several areas, such as emotional chatbots, sentiment analysis of comments in social media and healthcare intelligence, for understanding emotions in the conversation with emotion dynamics and generating emotionally coherent responses. ERC problem still remains a challenge. Both lexicon-based (Wu et al., 2006; Shaheen et al., 2014) and deep learning-based (Colnerič and Demšar, 2018) text emotion recognition methods that treat each utterance individually fail in this task as these works ignore some conversation-specific characteristics.



A good friend of mine passed away the other day. [sad]

Oh no, I'm so sorry. What happened? [sad]

Oh he had some pretty progressive cancer. [sad]

Did you know him well? [neutral]

Yeah, he was pretty awesome and supportive.

[happy] ✗ [sad] ✓

Figure 1: A dialogue from IEMOPCAP, in which the emotion of the last utterance by speaker A will be wrongly classified if the dialogue context is not taken into consideration.

In the past few years, recurrent neural network (RNN)-based solutions, such as CMN (Hazarika et al., 2018b), ICON (Hazarika et al., 2018a) and DialogueRNN (Majumder et al., 2019), have dominated this field due to the sequential nature of conversational context. Nonetheless, they share some inherent limitations: 1) RNN model performs poorly in grasping distant contextual information; 2) RNN-based methods are not capable of handling large-scale multiparty conversations.

With the rise of graph neural network (GNN) (Wu et al., 2020) in many natural language processing (NLP) tasks, researchers pay increasing attention to GNN-based ERC methods recently. Instead of modeling only sequential data recurrently in RNN, GNN is designed to capture all kinds of graph structure information via various aggregation algorithms. Existing GNN-based ERC methods, such as DialogueGCN (Ghosal et al., 2019), RGAT (Ishiwatari et al., 2020) and DAG-ERC (Shen et al., 2021), which are the state of the art, have demonstrated the superiority of GNN in modeling conversational structure information. A directed graph is constructed on each dialogue in these methods, where the nodes denote the individual utterances,

[*] Corresponding author.

Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 148–157
November 20–23, 2022. ©2022 Association for Computational Linguistics

and the edges indicate relationships between utterances. However, we notice that the relative position and speaker dependency information are mostly encoded together in one weight matrix according to the edge type in these methods, which can not exploit these conversation structure information sufficiently.

On the other hand, these methods do not work well on modeling speaker-specific context, which is also important in the ERC task. For example, in Figure 1 the third utterance spoken by speaker A is more influenced by speaker A's prior utterances rather than the second utterance spoken by speaker B, even though the latter is closer. Thus, in contextual modeling, we should consider both the emotional influence that speakers have on themselves during a conversation, i.e., self-speaker context, and context on the entire conversation flow, i.e., inter-speaker context, as well as the interaction between them.

In this paper, we propose a novel **S**peaker and **P**osition-**A**ware **G**NN model for **E**RC (S+PAGE) to settle the above drawbacks of existing methods. Our model contains three stages to fully consider both contextual modeling and conversation structure modeling. Specifically, given a sequence of utterances in the same dialogue, we first leverage a **T**wo-**S**tream **C**onversational **T**ransformer (TSCT) with the attentive masking mechanism to get both self and inter-speaker contextual features. Then, guided by the speaker dependency, we construct a conversation graph. We propose an enhanced relational graph convolution network (R-GCN), called SPGCN, to refine the contextual features with conversation structure information. Particularly, we introduce relational relative positional encoding in the aggregation algorithm to make SPGCN capable of capturing fine-grained positional information in a conversation. Finally, the global transfer of emotion labels is modeled by a conditional random field (CRF) layer with the features from both TSCT and SPGCN. Experimental results demonstrate the superiority of our model compared with state-of-the-art models. Ablation study illustrates the effectiveness of the proposed components in the model. To conclude, our contributions are as follows:

- We propose a new GNN-based ERC method, called S+PAGE, in which a novel graph neural network, namely SPGCN, is presented to better capture the conversation structure information.
- We present a two-stream conversational Transformer architecture to extract both self and inter-speaker contextual features.
- We conduct extensive experiments on four ERC benchmark datasets, and the results demonstrate that the proposed model achieves the competitive performance on all of them.

2 Related Works

2.1 Emotion Recognition in Conversation

Emotion recognition in conversation is a popular area in NLP. Many ERC datasets have been scripted and annotated in the past few years, such as IEMOCAP (Busso et al., 2008), MELD (Poria et al., 2018), DailyDialog (Li et al., 2017), EmotionLines (Chen et al., 2018) and EmoryNLP (Zahiri and Choi, 2018). IEMOCAP, MELD, and EmoryNLP are multimodal datasets, containing acoustic, visual and textual information, while the remaining two datasets are textual.

In recent years, ERC solutions are mostly deep learning-based models. CMN (Hazarika et al., 2018b) and ICON (Hazarika et al., 2018a) utilize gated recurrent unit (GRU) and memory networks to capture the dialogue dynamics. In IAAN (Yeh et al., 2019) and DialgueRNN (Majumder et al., 2019), attention mechanisms are applied to interact between the party state and global state. With the rise of Transformer and graph neural networks in NLP tasks, many works have also introduce them into the ERC task. (Zhong et al., 2019) propose KET, which is a structure of hierarchical Transformers assisted by external commonsense knowledge. DialogueXL (Shen et al., 2020) applies dialogue-aware self-attention to deal with the multi-party structures. In DialogueGCN (Ghosal et al., 2019) and RGAT (Ishiwatari et al., 2020), GCN (Kipf and Welling, 2016) and GAT (Veličković et al., 2017) are applied to refine the features with speaker dependencies and temporal information. DAG-ERC (Shen et al., 2021) applies a directed acyclic graph for conversation representation and it achieves the state-of-the-art performance on multiple ERC datasets.

2.2 Transformer

(Vaswani et al., 2017) first propose Transformer for machine translation task, whose success subsequently has been proved in various down-stream

NLP tasks. Self-attention mechanisms endow Transformer with the ability of capturing longer-range dependency among elements of an input sequence than the RNN structure. (Beltagy et al., 2020) propose a novel self-attention mechanism for feature extraction of long documents. Pre-trained models such as BERT (Devlin et al., 2018) and GPT (Brown et al., 2020) use Transformer encoder and decoder respectively to learn representations on large-scale datasets.

2.3 Graph Neural Network

Graph neural network has attracted a lot of attention in recent years, which learns a target node's representation by propagating neighbor information in the graph. (Kipf and Welling, 2016) propose a simple and well-behaved layer-wise propagation rule for neural network models and demonstrate its effectiveness in semi-supervised classification tasks. Better aggregation methods for large graphs are proposed in GAT (Veličković et al., 2017) and GraphSage (Hamilton et al., 2017). (Schlichtkrull et al., 2018) propose R-GCN to deal with the highly multi-relational data characteristic by assigning different aggregation structures for each relation type.

3 Methodology

The framework of our model is shown in Figure 2. We decompose the emotion classification procedure into three stages, i.e., contextual modeling, speaker dependency modeling, and global consistency modeling. In the first stage, we present a conversation-specific Transformer to get both self and inter-speaker contextual features. Then, a graph neural network is proposed to refine the features with conversation structure information, including the speaker dependency and relative position of each utterance. Subsequently, we employ conditional random field as the output layer to model the context of global consistency of emotion labels.

3.1 Problem Definition

The ERC task is to predict emotion labels (e.g., Happy, Sad, Neutral, Angry, Excited, and Frustrated) for utterances $\{u_1; u_2; \cdots ; u_N\}$, where N denotes the number of utterances in a conversation. Let S be the number of speakers in a given dataset. P is a mapping function, and $s = P(u_i)$ denotes utterance u_i uttered by speaker s, where $s \in \{1, \cdots , S\}$.

3.2 Utterance Encoding

Following previous works (Ghosal et al., 2019; Majumder et al., 2019), we use a simple architecture consisting of a single convolutional layer followed by a max-pooling layer and a fully connected layer to extract context-independent textual features of each utterance. The input of this network is the 300 dimensional pre-trained 840B GloVe vectors (Pennington et al., 2014). We use the output features, denoted as $\vec{u_i}$, as the representation of each utterance. Notice that we do not use any pre-trained model like BERT and RoBERTa to make utterance encoding for fairness of comparison with the baseline methods.

3.3 Contextual Modeling

We present a **T**wo-**S**tream **C**onversational **T**ransformer (TSCT) to better extract the contextual representation of each utterance in a conversation, which is also capable of handling multi-party conversations efficiently. The collection of utterance representations $U = \{\vec{u_1}; \vec{u_2}; \cdots ; \vec{u_N}\}$ is taken as the input. We design a multi-head self-attention mechanism, composed of two streams, i.e., the inter-speaker self-attention stream and the intra-speaker self-attention stream.

3.3.1 Inter-Speaker Self-Attention

The inter-speaker self-attention is same with the self-attention in vanilla Transformer, in which each utterance can attend to all positions in the dialogue as shown in Figure 3(a). It is calculated as:

$$q_i^t, k_i^t, v_i^t = h_i^{t-1} W_{iq}^t, h_i^{t-1} W_{ik}^t, h_i^{t-1} W_{iv}^t \tag{1}$$

$$z_i^t = softmax(\frac{q_i^t (k_i^t)^T}{\sqrt{d}}) v_i^t \tag{2}$$

where W_{iq}^t, W_{ik}^t and W_{iv}^t are three learnable weight matrices for attention head i at layer t.

3.3.2 Intra-Speaker Self-Attention

The intra-speaker self-attention models speaker-specific contextual information by only computing attention on the same speaker's utterances in a dialogue. In this way, the model is able to capture the emotional influence that speakers have on themselves during the conversation. It is implemented by the attentive masking strategy as illustrated in Figure 3(b) and formulated as:

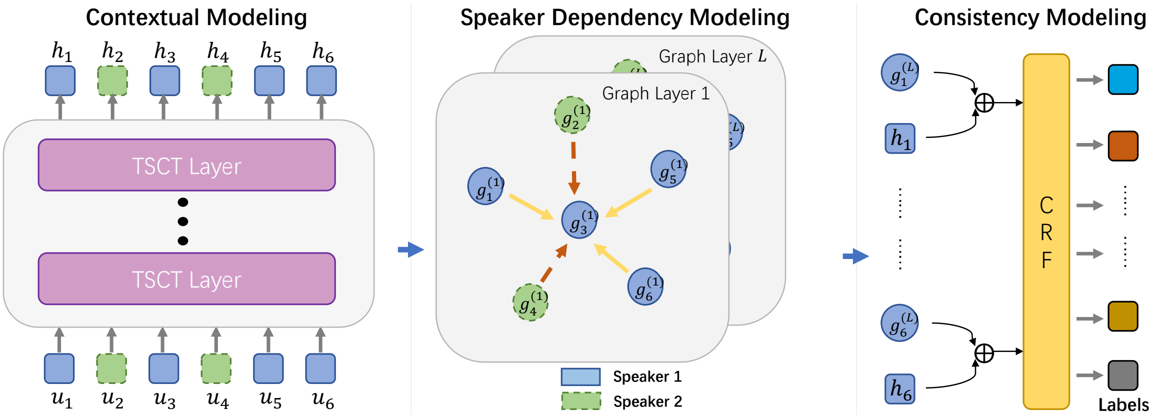

Contextual Modeling
h_1 h_2 h_3 h_4 h_5 h_6
TSCT Layer
TSCT Layer
u_1 u_2 u_3 u_4 u_5 u_6
Speaker Dependency Modeling
Graph Layer L
Graph Layer 1
$g_1^{(1)}$ $g_2^{(1)}$ $g_3^{(1)}$ $g_4^{(1)}$ $g_5^{(1)}$ $g_6^{(1)}$
Speaker 1
Speaker 2
Consistency Modeling
$g_1^{(L)}$ h_1 $g_6^{(L)}$ h_6
CRF
Labels

Figure 2: The overall framework of S+PAGE. First, contextualized representation of each utterance is obtained by contextual modeling part. Subsequently, we employ SPGCN to model the speaker dependency and position information. Finally, the CRF layer applied to model the consistency using information from the previous parts. ⊕ denotes the concatenation operation. L is the total number of graph layers.

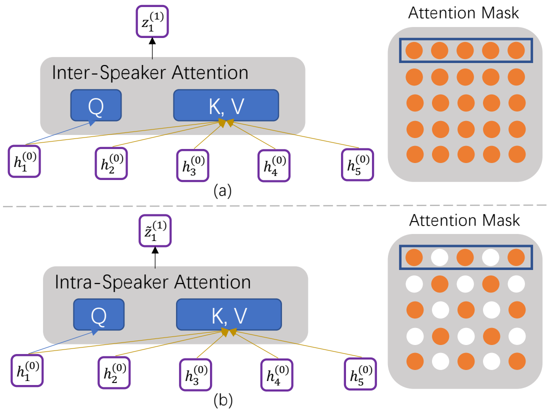


Figure 3: (a) Inter-speaker self-attention: the attention among all speakers, same with vanilla Transformer.(b) Intra-speaker self-attention: the attention only on the utterances spoke by the current speaker.

$$\tilde{z}_i^t = softmax(\frac{q_i^t(k_i^t)^T}{\sqrt{d}} + m)v_i^t \tag{3}$$

where $m \in \mathbb{R}^{N \times N}$ is the attentive masking matrix. The elements of m are set as below:

$$m_{ij} = \begin{cases} -\infty & P(u_i) \neq P(u_j) \\ 0 & otherwise \end{cases} \tag{4}$$

where $P(\cdot)$ is the function that maps the utterance and its corresponding speaker.

Each attention head i of the t-th layer in TSCT, denoted as $head_i^t$, is the concatenation of the z_i and $\tilde{z}_i$, and the output of the multi-head attention can be formulated as follows:

$$MultiHead_i^t = \|_{i=1}^{M} head_i^t \tag{5}$$

where $\|$ denotes concatenation operation. M is the number of attention heads, while $1 \leq i \leq M$.

Following the structure of the original Transformer, the output of the TSCT layer can be generated by passing $MultiHead_i^t$ through a FF (feed-forward network):

$$h^t = \text{LayerNorm}(\text{FF}(MultiHead_i^t)) \tag{6}$$

3.4 Speaker Dependency Modeling

After extracting the contextual features, we introduce a novel graph neural network, named SPGCN, to propagate structure-aware utterance features. Specifically, in SPGCN, speaker dependency and position information are modeled by edge types and edge weights respectively, and are combined in the aggregation function to update the features.

3.4.1 SPGCN

Graph Architecture We construct a directed graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R}, \mathcal{W})$, for each dialogue with N utterances. The nodes in the graph are the utterances in the conversation, i.e., $V = \{v_1; v_2; \cdots, v_N\}$. $(v_i, v_j, r_{ij}) \in \mathcal{E}$ denotes a labeled edge (relation), where $r_{ij} \in \mathcal{R}$ is a relation type, defined according to speaker identity and relative distance. $\mathcal{W}$ represents the set of edge weights.

Nodes Feature vector g_i of each node v_i is initialized as the output of the TSCT layer, i.e., h_i. g_i is modified by the aggregation algorithm through the stacked graphical layers in GNN. The output feature is described as g_i^l, where l denotes the number of layers.

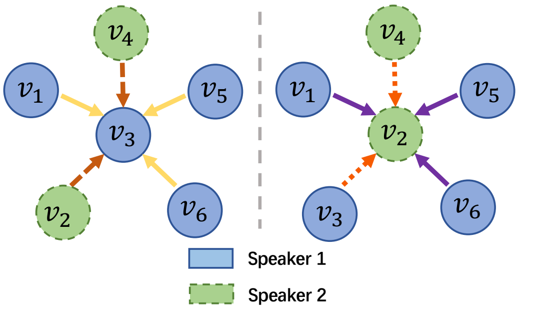


Figure 4: An example of incoming edges for nodes v_3 (left) and v_2 (right) in the dialogue graph. Different types of arrows denote different edge types. Nodes share the same edge types if they are spoke by the same speaker. v_3, v_1 and v_5 are spoke by speaker1, thus the edge between v_3, v_1 and the edge between v_3, v_5 belong to the same edge type.

Edges Instead of only focusing on past utterances, we take converse influence into account (Ghosal et al., 2019). We construct edges $\mathcal{E}$ with a sliding window for each utterance. The window sizes p and f denote the number of past and future utterances from the target utterance. Each utterance node v_i has an edge with p utterances of the past: $\{v_{i-1}, v_{i-2}, ..., v_{i-p}\}$, f utterances of the future: $\{v_{i+1}, v_{i+2}, ..., v_{i+f}\}$, and itself.

Edge Types The relation type $r \in R$ is determined by the *speaker identity*. Assuming there are S distinct speakers in a dialogue, there should be $N_e = S^2$ relation types in the constructed graph $\mathcal{G}$. Two utterances share the same edge type only if they are uttered by the same speaker. For example, in Figure 4 the incoming edges $v_1 \to v_3$ and $v_5 \to v_3$ share the same edge type, and $v_4 \to v_3$ is a different edge type.

Edge Weights Edge weight $\alpha_{ij} \in \mathcal{W}$ is computed by an attention mechanism. The particular attentional setup in our model closely follows the work of GAT (Veličković et al., 2017). The input of the attention module is a set of node features from the last layer. Motivated by (Shaw et al., 2018), which shows that absolute positional encoding is not effective for the model to capture the information of relative word order, we inject relative positional encoding into the attention mechanism.

$$\beta_{ij} = E_p(o(v_j) - o(v_i)) \tag{7}$$

$$\Gamma_{ij} = LReLU\left(\vec{a}^T\left[Wg_i^{l-1}\|(Wg_j^{l-1} + \beta_{ij})\right]\right) \tag{8}$$

$$\alpha_{ij} = \frac{\exp \Gamma_{ij}}{\sum_{k\in Ni} \exp \Gamma_{ik}} \tag{9}$$

β_{ij} denotes the signed relative position representation between utterance i and utterance j in a dialogue, which is encoded by a trainable embedding matrix E_p. $o(\cdot)$ is a mapping function between utterance and its absolute position in the dialogue sequence. $LReLU$ denotes the activation function $LeakyReLU$. W is a weight matrix applied to every node. N_i is the number of nodes linked with node i. $\vec{a}$ is a parametrized weight vector. $\cdot^T$ represents transposition, and $\|$ is the concatenation operation.

Aggregation Function Inspired by R-GCN (Schlichtkrull et al., 2018), we define the following aggregation algorithm to calculate the forward-pass update of a node in the graph:

$$\widetilde{g_i}^l = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in N_i^r}\frac{\alpha_{ij}^l}{c_{i,r}}W_r^l g_i^{l-1} + \alpha_{ii}^l W_o^l g_i^{l-1}\right) \tag{10}$$

where $\widetilde{g_i}^l$ is the aggregated state of node i in the l-th layer. N_i^r denotes the set of neighbors of utterance i under the edge type $r \in R$. $c_{i,r}$ is a normalization constant, and we set $c_{i,r} = |N_i^r|$ in our experiment. W_r^l and W_o^l are learnable weight matrices, and $\sigma(\cdot)$ is an activation function, such as the ReLU. Different from R-GCN, we use edge weights calculated by Equation 9 to involve fine-grained positional information in a conversation.

After the aggregation, we employ a gate fusion function to make $\widetilde{g_i}^l$ interact with its hidden state at the previous layer. Finally, the representation at the l-th layer is formulated as:

$$g^{'} = [\widetilde{g_i}^l; g_i^{l-1}; \widetilde{g_i}^l * g_i^{l-1}; \widetilde{g_i}^l - g_i^{l-1}] \tag{11}$$

$$\epsilon = sigmoid\left(W_f g^{'} + b_f\right) \tag{12}$$

$$g_i^l = \epsilon * \widetilde{g_i}^l + (1 - \epsilon) * g_i^{l-1} \tag{13}$$

where $l \geq 1$, and W_f and b_f are trainable parameters. $g^{'}$ is the concatenation of the four vectors.

3.5 Consistency Modeling

Instead of directly using a softmax function in the output layer, we employ conditional random field (CRF) to yield final emotion tags of each utterance.

Our motivation is to model the emotional consistency in a conversation, i.e., the emotion transfer. Using the CRF layer enables the model to take into account the dependency between emotion tags in neighborhoods and choose the globally best tag sequence for the entire conversation at once.

Following the describe by Lample et al., for an input set of utterances $U = \{u_1, u_2, ..., u_N\}$ and a sequence of tag predictions $y = \{y_1, y_2, .., y_N\}$, $y_i \in 1, \cdots, K$ (K is number of emotion tags), the score of the sequence is defined as,

$$score(\mathbf{U}, \mathbf{y}) = \sum_{i=0}^{n} D_{y_i, y_{i+1}} + \sum_{i=1}^{n} B_{i, y_i} \qquad (14)$$

where $D \in \mathbb{R}^{K \times K}$ is the matrix of transition, $B \in \mathbb{R}^n \times K$ is the output score of the prepended classification model. The model is trained to maximize the log-probability of the correct tag sequence:

$$\log(p(\mathbf{y} \mid \mathbf{U})) = score(\mathbf{U}, \mathbf{y}) - \log\left(\sum_{\tilde{\mathbf{y}} \in \mathbf{Y}} e^{score(\mathbf{U}, \tilde{\mathbf{y}})}\right) \qquad (15)$$

where Y is set of all possible tag sequences. Equation 15 is computed using dynamic programming, while Viterbi applied applied to get most likely sequence following the work of Rabiner et al. (Rabiner, 1989).

4 Experiments

4.1 Datasets and Baselines

We evaluate our S+PAGE model on four widely-used benchmark datasets – **IEMOCAP** (Busso et al., 2008), which is a audiovisual dataset consisting of dyadic conversations where actors perform improvisations or scripted scenarios, **MELD** (Poria et al., 2018) and **EmoryNLP** (Zahiri and Choi, 2018), both of which are multi-modal and multi-party datasets created from scripts of the Friends TV series, and **DailyDialog** (Li et al., 2017), which is a human-written dyadic dataset covering various topics about our daily life. For this work, we only consider emotion recognition based on textual features, and thus some recent ERC solutions on multi-modal features (Chudasama et al., 2022; Hu et al., 2022) are not selected as our baselines for fairness. The statistic of them is shown in Table 1.

Dataset	# Conversations			# Uterrances		
	Train	Val	Test	Train	Val	Test
IEMOCAP	120		31	5810		1623
MELD	1038	114	280	9989	1109	2610
DailyDialog	11118	1000	1000	87170	8069	7740
EmoryNLP	713	99	85	9934	1344	1328

Table 1: The statistics of the datasets.

For a comprehensive performance evaluation, we choose **CNN**, **CNN+cLSTM** (Poria et al., 2017), **DialogueRNN** (Majumder et al., 2019) as baselines of CNN and RNN-based methods, **KET** (Zhong et al., 2019) as advanced Transformer-based approach with external commonsense knowledge included, **DialogueGCN** (Ghosal et al., 2019), **RGAT** (Ishiwatari et al., 2020) and **DAG-ERC** (Shen et al., 2021) as GNN-based approaches. Particularly, these three GNN-based models are the recent state of the art. DialogueGCN applies GCN to model speaker dependency, but it does not contain fine-grained positional information. Similarly, DAG-ERC applies a directed acyclic graph for conversation representation, which lack positional information in a conversation too. RGAT encodes both speaker dependency and relative positional encoding into the edge type, and use graph attention networks to make information aggregation.

For the evaluation metrics, we choose micro-averaged F1 for DailyDialog and weighted-average F1 for the other datasets, following previous works (Ishiwatari et al., 2020; Shen et al., 2021).

4.2 Experimental Settings

We set the initial learning rate as 1e-4 in the Transformer layers, 2e-4 in the SPGCN layers and 2e-2 in the CRF layer. AdamW optimizer is used under a scheduled learning rate following (Vaswani et al., 2017). The number of dimensions of the utterance representations and contextual embeddings is set to 300. We set the layer number of TSCT and SPGCN to 8 and 3 respectively. We set the dropout rate and number of attention head in TSCT to be 0.1 and 8 respectively. 3-head attention is used during calculating the edge weights. We also conduct experiments with different window sizes and SPGCN layers. We choose the hyper-parameters that achieve the best score on each dataset by using development data. The training and testing process is run on a single Tesla P100 GPU with 32G memory. The reported results of our implemented models are all based on the average score

Model	IEMOCAP	MELD	DailyDialog	EmoryNLP
CNN	48.18	55.86	49.34	32.59
CNN+cLSTM	54.95	56.87	50.24	32.89
DialogueRNN	62.75	57.03	-	-
KET	59.56	58.18	53.37	33.95
DialogueGCN	64.18	58.10	-	-
RGAT	65.22	60.91	54.31	34.42
DAG-ERC	68.03	63.65	59.33	39.02
S+PAGE	68.75 (0.11)	63.43 (0.15)	64.08 (0.21)	39.16 (0.12)
S+PAGE$_{Bert}$	68.77 (0.13)	63.25 (0.18)	**64.18** (0.25)	38.96 (0.13)
S+PAGE$_{RoBERTa}$	**68.93** (0.12)	**64.67** (0.15)	64.11 (0.21)	**40.05** (0.14)

Table 2: Overall performance on the four datasets.

of 5 random runs on the test sets.

5 Results and Analysis

5.1 Overall Performance

We compare our model with the baseline methods, and the results are reported in Table 2. We can note that our proposed S+PAGE has the best performance on all the four benchmark datasets. All GNN-based models outperform RNN-based models, which indicates the necessity of modeling the conversation structure information in the ERC task. Compared with existing GNN-based models, our model even has competitive results. There are three main advantages that contribute to our performance: 1) contextual modeling with both self and inter-speaker dependency, 2) a better speaker dependency and relative positional encoding in GNN, 3) consistency modeling of global emotion transfer.

We find that the improvements on MELD and EmoryNLP are not significant without utilizing pre-trained language models, i.e, BERT and RoBERTa. The performances of S+PAGE enhanced after replacing GloVe vectors by embeddings from pre-trained language models. This is because both datasets consturcted on Friends TV series, extra knowledge from large pre-trained language help the model to understand the dialogue better.

5.2 Ablation Study

To better understand the contribution of each component in our proposed model, we conduct experiments by replacing TSCT with the vanilla Transformer, and removing SPGCN and CRF from our model respectively. The results on IEMOCAP and MELD are shown in Table 3. We can observe that when TSCT is removed, the weighted F1 score drops more on MELD than that on IEMOCAP. This shows the superiority of TSCT on contextual feature extraction of multi-party conversations, as there are more speakers in dialogues of MELD. Removal of SPGCN leads to significant drop on both datasets, which implies the importance of SPGCN to refine the contextual features with speaker dependency and relative position. Meanwhile, after removing CRF layer, we can also observe the performance degradation. It implies that the modeling of label consistency is essential in the ERC task. To sum up, all of the three components contribute to the performance improvement of S+PAGE.

Method	IEMOCAP	MELD
S+PAGE	68.93	64.67
- TSCT	68.11 (↓0.82)	63.21 (↓1.46)
- SPGCN	64.25 (↓4.68)	62.03 (↓2.64)
- CRF	68.29 (↓0.64)	64.24 (↓0.43)

Table 3: Results of ablation study.

5.3 Whether SPGCN outperforms other graph structures?

We conduct experiments on IEMOCAP by replacing SPGCN with the graph structures in DialogueGCN, RGAT and DAG-ERC respectively. As shown in Table 4, S+PAGE still outperforms the other methods significantly. Notice that both DialogueGCN and RGAT with our contextual and consistency modeling perform better than their original versions. This indicates the necessary of the speaker-spcific information modeling in contextual modeling and speaker emotional consistency modeling, which is neglected in the previous methods. We use language embeddings from BERT$_{\text{base}}$ in RGAT and RoBERTa$_{\text{large}}$ in DAG follow the original papers for fair comparision.

5.4 Effect of Window Size

We analyze the influence of past and future window sizes by conducting experiments with window size w of $(4, 4)$, $(6, 6)$, $(8, 8)$, $(10, 10)$, $(20, 20)$, $(30, 30)$, $(40, 40)$ on IEMOCAP dataset. As shown in Figure 5, the F1 score of S+PAGE, RGAT and DialogueGCN significantly increase, when the window sizes expand from 4 to 10. The reason is that useful contextual information keeps

Method	IEMOCAP
S+PAGE	68.93
S+PAGE(-SPGCN) + GCN	64.82
S+PAGE(-SPGCN) + RGAT	65.78
S+PAGE(-SPGCN) + DAG	67.93

Table 4: Results of replacing SPGCN with other graph structures.

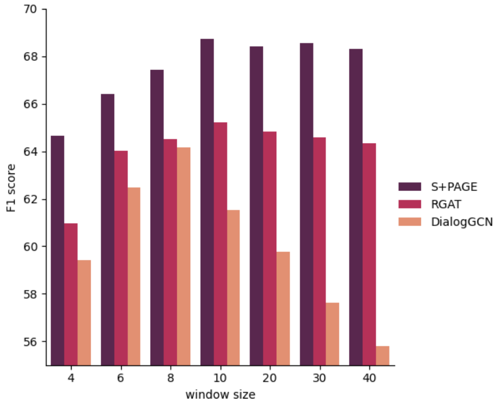


Figure 5: Results of varying window sizes.

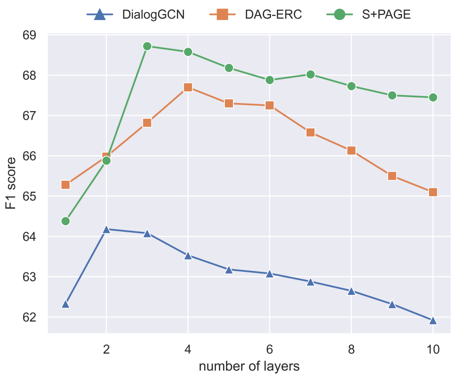


Figure 6: Graph layer ablation

growing with the increasing of w. However, after window sizes exceed 20, the F1 score drops for both DialogueGCN and RGAT. The reason is that the amount of useless long-range dependency increases when the window size continuously grows, which hinders the models from efficiently capturing crucial context. In contrast, the performance of S+PAGE fluctuates in a relatively narrow range, which shows the robustness of our model on varied window sizes. We can infer that the relative positional encoding endows capability of distinguishing critical contextual information to our model.

5.5 Number of SPGCN layers

We further explore the relationship between model performance and the number of layers of the SPGCN. Stacking too many layers of GNN may lead to performance degradation because of over-smoothing problem (Kipf and Welling, 2016). As shown in Figure 6, we conduct an experiment on IEMOCAP by setting different number of layers of the SPGCN, with the comparison of DiaglogGCN and DAG-ERC. As can be seen from Figure 6, DialogGCN suffers from a significant performance degradation after number of layers exceeds 3. On the other hand, for SPGCN and DAG, the drop seems to be more slight, which indicate the over-smooth problem alleviated in both structures.

Method	IEMOCAP
S+PAGE(RPE)	68.93
S+PAGE(APE)	66.38
S+PAGE(PER)	65.93

Table 5: Results of S+PAGE with other positional encoding methods in SPGCN. RPE is proposed relative positional embedding, APE is absolute positional embedding and PER is positional embeddings in RGAT.

5.6 Effect of Relative Positional Embedding

In this part, we conduct experiments to study whether our relative positional embedding(REP) in SPGCN is superior to other positional embedding methods. We replace REP with the popular absolute positional embedding (APE) and the position encoding (PE) implemented in RGAT. As shown in Table 5, the model with our RPE significantly outperforms the models with other position embedding methods.

6 Conclusion

In this paper, we propose a novel graph neural network-based model, named S+PAGE, for emotion recognition in conversation (ERC). Specifically, S+PAGE contains three parts, i.e., contextual modeling, speaker dependency modeling, and consistency modeling. In contextual modeling, we present a new Transformer structure with two-stream attention mechanism to better capture the self and inter-speaker contextual features. In speaker dependency modeling, we introduce a novel GNN model, named SPGCN, to refine the features with the conversation structure information including speaker dependency and relative position information. Furthermore, we use a CRF layer to model emotion transfer in the consistency modeling part. Experimental results on four ERC benchmark datasets demonstrate the superiority of our model.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

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.

Sheng-Yeh Chen, Chao-Chun Hsu, Chuan-Chun Kuo, Lun-Wei Ku, et al. 2018. Emotionlines: An emotion corpus of multi-party conversations. *arXiv preprint arXiv:1802.08379*.

Vishal Chudasama, Purbayan Kar, Ashish Gudmalwar, Nirmesh Shah, Pankaj Wasnik, and Naoyuki Onoe. 2022. M2fnet: Multi-modal fusion network for emotion recognition in conversation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4652–4661.

Niko Colnerič and Janez Demšar. 2018. Emotion recognition on twitter: Comparative study and training a unison model. *IEEE transactions on affective computing*, 11(3):433–446.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Deepanway Ghosal, Navonil Majumder, Soujanya Poria, Niyati Chhaya, and Alexander Gelbukh. 2019. Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation. *arXiv preprint arXiv:1908.11540*.

William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1025–1035.

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.

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 conference. Association for Computational Linguistics. North American Chapter. Meeting*, volume 2018, page 2122. NIH Public Access.

Dou Hu, Xiaolong Hou, Lingwei Wei, Lianxin Jiang, and Yang Mo. 2022. Mm-dfn: Multimodal dynamic fusion network for emotion recognition in conversations. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7037–7041. IEEE.

Taichi Ishiwatari, Yuki Yasuda, Taro Miyazaki, and Jun Goto. 2020. Relation-aware graph attention networks with relational position encodings for emotion recognition in conversations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7360–7370.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander Gelbukh, and Erik Cambria. 2019. Dialoguernn: An attentive rnn for emotion detection in conversations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6818–6825.

Jeffrey Pennington, Richard Socher, and Christopher D 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.

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.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. 2018. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *arXiv preprint arXiv:1810.02508*.

Lawrence R Rabiner. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer.

Shadi Shaheen, Wassim El-Hajj, Hazem Hajj, and Shady Elbassuoni. 2014. Emotion recognition from text based on automatically generated rules. In *2014 IEEE International Conference on Data Mining Workshop*, pages 383–392. IEEE.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Weizhou Shen, Junqing Chen, Xiaojun Quan, and Zhixian Xie. 2020. Dialogxl: All-in-one xlnet for multi-party conversation emotion recognition. *arXiv preprint arXiv:2012.08695*.

Weizhou Shen, Siyue Wu, Yunyi Yang, and Xiaojun Quan. 2021. Directed acyclic graph network for conversational emotion recognition. *arXiv preprint arXiv:2105.12907*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Chung-Hsien Wu, Ze-Jing Chuang, and Yu-Chung Lin. 2006. Emotion recognition from text using semantic labels and separable mixture models. *ACM transactions on Asian language information processing (TALIP)*, 5(2):165–183.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24.

Sung-Lin Yeh, Yun-Shao Lin, and Chi-Chun Lee. 2019. An interaction-aware attention network for speech emotion recognition in spoken dialogs. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6685–6689. IEEE.

Sayyed M Zahiri and Jinho D Choi. 2018. Emotion detection on tv show transcripts with sequence-based convolutional neural networks. In *Workshops at the thirty-second aaai conference on artificial intelligence*.

Peixiang Zhong, Di Wang, and Chunyan Miao. 2019. Knowledge-enriched transformer for emotion detection in textual conversations. *arXiv preprint arXiv:1909.10681*.