Directed Acyclic Graph Network for Conversational Emotion Recognition

Weizhou Shen, Siyue Wu, Yunyi Yang, Xiaojun Quan[*]
School of Computer Science and Engineering, Sun Yat-sen University, China
{shenwzh3, wusy39, yangyy37}@mail2.sysu.edu.cn
quanxj3@mail.sysu.edu.cn

Abstract

The modeling of conversational context plays a vital role in emotion recognition from conversation (ERC). In this paper, we put forward a novel idea of encoding the utterances with a directed acyclic graph (DAG) to better model the intrinsic structure within a conversation, and design a directed acyclic neural network, namely DAG-ERC[1], to implement this idea. In an attempt to combine the strengths of conventional graph-based neural models and recurrence-based neural models, DAG-ERC provides a more intuitive way to model the information flow between long-distance conversation background and nearby context. Extensive experiments are conducted on four ERC benchmarks with state-of-the-art models employed as baselines for comparison. The empirical results demonstrate the superiority of this new model and confirm the motivation of the directed acyclic graph architecture for ERC.

1 Introduction

Utterance-level emotion recognition in conversation (ERC) is an emerging task that aims to identify the emotion of each utterance in a conversation. This task has been recently concerned by a considerable number of NLP researchers due to its potential applications in several areas, such as opinion mining in social media (Chatterjee et al., 2019) and building an emotional and empathetic dialog system (Majumder et al., 2020).

The emotion of a query utterance is likely to be influenced by many factors such as the utterances spoken by the same speaker and the surrounding conversation context. Indeed, how to model the conversational context lies at the heart of this task (Poria et al., 2019a). Empirical evidence also shows

---
[*]Corresponding author.
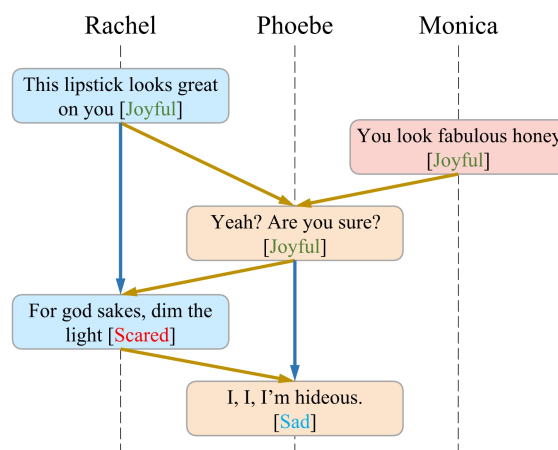[1]The code is available at https://github.com/shenwzh3/DAG-ERC

Figure 1: Conversation as a directed acyclic graph, with brown directed edges representing the information propagation between speakers and blue ones representing the information propagation inside a same speaker.

that a good representation of conversation context significantly contributes to the model performance, especially when the content of query utterance is too short to be identified alone (Ghosal et al., 2019).

Numerous efforts have been devoted to the modeling of conversation context. Basically, they can be divided into two categories: graph-based methods (Zhang et al., 2019a; Ghosal et al., 2019; Zhong et al., 2019; Ishiwatari et al., 2020; Shen et al., 2020) and recurrence-based methods (Hazarika et al., 2018a; Hazarika et al., 2018b; Majumder et al., 2019; Ghosal et al., 2020). For the graph-based methods, they concurrently gather information of the surrounding utterances within a certain window, while neglecting the distant utterances and the sequential information. For the recurrence-based methods, they consider the distant utterances and sequential information by encoding the utterances temporally. However, they tend to update the query utterance's state with only relatively limited information from the nearest utterances, making them difficult to get a satisfying performance.

1551

According to the above analysis, an intuitively better way to solve ERC is to allow the advantages of graph-based methods and recurrence-based models to complement each other. This can be achieved by regarding each conversation as a directed acyclic graph (DAG). As illustrated in Figure 1, each utterance in a conversation only receives information from some previous utterances and cannot propagate information backward to itself and its predecessors through any path. This characteristic indicates that a conversation can be regarded as a DAG. Moreover, by the information flow from predecessors to successors through edges, DAG can gather information for a query utterance from both the neighboring utterances and the remote utterances, which acts like a combination of graph structure and recurrence structure. Thus, we speculate that DAG is a more appropriate and reasonable way than graph-based structure and recurrence-based structure to model the conversation context in ERC.

In this paper, we propose a method to model the conversation context in the form of DAG. Firstly, rather than simply connecting each utterance with a fixed number of its surrounding utterances to build a graph, we propose a new way to build a DAG from the conversation with constraints on speaker identity and positional relations. Secondly, inspired by DAGNN (Thost and Chen, 2021), we propose a directed acyclic graph neural network for ERC, namely DAG-ERC. Unlike the traditional graph neural networks such as GCN (Kipf and Welling, 2016) and GAT (Veličković et al., 2017) that aggregate information from the previous layer, DAG-ERC can recurrently gather information of predecessors for every utterance in a single layer, which enables the model to encode the remote context without having to stack too many layers. Besides, in order to be more applicable to the ERC task, our DAG-ERC has two improvements over DAGNN: (1) a relation-aware feature transformation to gather information based on speaker identity and (2) a contextual information unit to enhance the information of historical context. We conduct extensive experiments on four ERC benchmarks and the results show that the proposed DAG-ERC achieves comparable performance with the state-of-the-art models. Furthermore, several studies are conducted to explore the effect of the proposed DAG structure and the modules of DAG-ERC.

The contributions of this paper are threefold. First, we are the first to consider a conversation as a directed acyclic graph in the ERC task. Second, we propose a method to build a DAG from a conversation with constraints based on the speaker identity and positional relations. Third, we propose a directed acyclic graph neural network for ERC, which takes DAGNN as its backbone and has two main improvements designed specifically for ERC.

2 Related work

2.1 Emotion Recognition in Conversation

Recently, several ERC datasets with textual data have been released (Busso et al., 2008; Schuller et al., 2012; Zahiri and Choi, 2017; Li et al., 2017; Chen et al., 2018; Poria et al., 2019b), arousing the widespread interest of NLP researchers. In the following paragraphs, we divide the related works into two categories according to the methods they use to model the conversation context.

Graph-based Models DialogGCN (Ghosal et al., 2019) treats each dialog as a graph in which each utterance is connected with the surrounding utterances. RGAT (Ishiwatari et al., 2020) adds positional encodings to DialogGCN. ConGCN (Zhang et al., 2019a) regards both speakers and utterances as graph nodes and makes the whole ERC dataset a single graph. KET (Zhong et al., 2019) uses hierarchical Transformers (Vaswani et al., 2017) with external knowledge. DialogXL (Shen et al., 2020) improves XLNet (Yang et al., 2019) with enhanced memory and dialog-aware self-attention.[2]

Recurrence-based Models In this category, ICON (Hazarika et al., 2018a) and CMN (Hazarika et al., 2018b) both utilize gated recurrent unit (GRU) and memory networks. HiGRU (Jiao et al., 2019) contains two GRUs, one for utterance encoder and the other for conversation encoder. DialogRNN (Majumder et al., 2019) is a recurrence-based method that models dialog dynamics with several RNNs. COSMIC (Ghosal et al., 2020) is the latest model, which adopts a network structure very close to DialogRNN and adds external commonsense knowledge to improve performance.

2.2 Directed Acyclic Graph Neural Network

Directed acyclic graph is a special type of graph structure that can be seen in multiple areas, for example, the parsing results of source code (Allamanis et al., 2018) and logical formulas (Crouse

---

[2]We regard KET and DialogXL as graph-based models because they both adopt Transformer in which self-attention can be viewed as a fully-connected graph in some sense.

et al., 2019). A number of neural networks that employ DAG architecture have been proposed, such as Tree-LSTM (Tai et al., 2015), DAG-RNN(Shuai et al., 2016), D-VAE (Zhang et al., 2019b), and DAGNN (Thost and Chen, 2021). DAGNN is different from the previous DAG models in the model structure. Specifically, DAGNN allows multiple layers to be stacked, while the others have only one single layer. Besides, instead of merely carrying out naive sum or element-wise product on the predecessors' representations, DAGNN conducts information aggregation using graph attention.

3 Methodology

3.1 Problem Definition

In ERC, a conversation is defined as a sequence of utterances $\{u_1, u_2, ..., u_N\}$, where N is the number of utterances. Each utterance u_i consists of n_i tokens, namely $u_i = \{w_{i1}, w_{i2}, ..., w_{in_i}\}$. A discrete value $y_i \in \mathcal{S}$ is used to denote the emotion label of u_i, where $\mathcal{S}$ is the set of emotion labels. The speaker identity is denoted by a function $p(\cdot)$. For example, $p(u_i) \in \mathcal{P}$ denotes the speaker of u_i and $\mathcal{P}$ is the collection of all speaker roles in an ERC dataset. The objective of this task is to predict the emotion label y_t for a given query utterance u_t based on dialog context $\{u_1, u_2, ..., u_N\}$ and the corresponding speaker identity.

3.2 Building a DAG from a Conversation

We design a directed acyclic graph (DAG) to model the information propagation in a conversation. A DAG is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$. In this paper, the nodes in the DAG are the utterances in the conversation, i.e., $\mathcal{V} = \{u_1, u_2, ..., u_N\}$, and the edge $(i, j, r_{ij}) \in \mathcal{E}$ represents the information propagated from u_i to u_j, where $r_{ij} \in \mathcal{R}$ is the relation type of the edge. The set of relation types of edges, $\mathcal{R} = \{0, 1\}$, contains two types of relation: 1 for that the two connected utterances are spoken by the same speaker, and 0 for otherwise.

We impose three constraints to decide when an utterance would propagate information to another, i.e., when two utterances are connected in the DAG:
Direction: $\forall j > i, (j, i, r_{ji}) \notin \mathcal{E}$. A previous utterance can pass message to a future utterance, but a future utterance cannot pass message backwards.
Remote information: $\exists \tau < i, p(u_\tau) = p(u_i), (\tau, i, r_{\tau i}) \in \mathcal{E}$ and $\forall j < \tau, (j, i, r_{ji}) \notin \mathcal{E}$. For each utterance u_i except the first one, there is a previous utterance u_τ that is spoken by the same speaker as

---

Algorithm 1 Building a DAG from a Conversation

Input: the dialog $\{u_1, u_2, ..., u_N\}$, speaker identity $p(\cdot)$, hyper-parameter ω
Output: $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$
1: $\mathcal{V} \leftarrow \{u_1, u_2, ..., u_N\}$
2: $\mathcal{E} \leftarrow \emptyset$
3: $\mathcal{R} \leftarrow \{0, 1\}$
4: **for all** $i \in \{2, 3, ..., N\}$ **do**
5: $\quad c \leftarrow 0$
6: $\quad \tau \leftarrow i - 1$
7: $\quad$ **while** $\tau > 0$ and $c < \omega$ **do**
8: $\quad\quad$ **if** $p(u_\tau) = p(u_i)$ **then**
9: $\quad\quad\quad \mathcal{E} \leftarrow \mathcal{E} \cup \{(\tau, i, 1)\}$
10: $\quad\quad\quad c \leftarrow c + 1$
11: $\quad\quad$ **else**
12: $\quad\quad\quad \mathcal{E} \leftarrow \mathcal{E} \cup \{(\tau, i, 0)\}$
13: $\quad\quad$ **end if**
14: $\quad\quad \tau \leftarrow \tau - 1$
15: $\quad$ **end while**
16: **end for**
17: return $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$

---

u_i. The information generated before u_τ is called remote information, which is relatively less important. We assume that when the speaker speaks u_τ, she/he has been aware of the remote information before u_τ. That means, u_τ has included the remote information and it will be responsible for propagating the remote information to u_i.

Local information: $\forall l, \tau < l < i, (l, i, r_{li}) \in \mathcal{E}$. Usually, the information of the local context is important. Consider u_τ and u_i defined in the second constraint. We assume that every utterance u_l in between u_τ and u_i contains local information, and they will propagate the local information to u_i.

The first constraint ensures the conversation to be a DAG, and the second and third constraints indicate that u_τ is the cut-off point of remote and local information. We regard u_τ as the ω-th latest utterance spoken by $p(u_i)$ before u_i, where ω is a hyper-parameter. Then for each utterance u_l in between u_τ and u_i, we make a directed edge from u_l to u_i. We show the above process of building a DAG in Algorithm 1.

An example of the DAG is shown in Figure 2. In general, our DAG has two main advancements compared to the graph structures developed in previous works (Ghosal et al., 2019; Ishiwatari et al., 2020): First, our DAG doesn't have edges from future utterances to previous utterances, which we
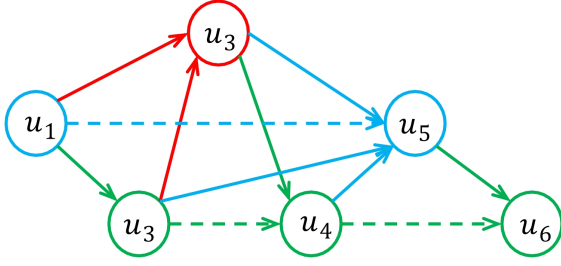
1553

Figure 2: An example DAG built from a three-party conversation, with $\omega = 1$. The three speakers' utterances are colored by red, blue and green, respectively. Solid lines represent the edges of local information, and dash lines denote the edges of remote information.

argue is more reasonable and realistic, as the emotion of a query utterance should not be influenced by the future utterances in practice. Second, our DAG seeks a more meaningful u_τ for each utterance, rather than simply connecting each utterance with a fixed number of surrounding utterances.

3.3 Directed Acyclic Graph Neural Network

In this section, we introduce the proposed **D**irected **A**cyclic **G**raph Neural Network for **ERC** (DAG-ERC). The framework is shown in Figure 3.

3.3.1 Utterance Feature Extraction

DAG-ERC regards each utterance as a graph node, the feature of which can be extracted by a pre-trained Transformer-based language model. Following the convention, the pre-trained language model is firstly fine-tuned on each ERC dataset, and its parameters are then frozen while training DAG-ERC. Following Ghosal et al. (2020), we employ RoBERTa-Large (Liu et al., 2019), which has the same architecture as BERT-Large (Devlin et al., 2018), as our feature extractor. More specifically, for each utterance u_i, we prepend a special token $[CLS]$ to its tokens, making the input a form of $\{[CLS], w_{i1}, w_{i2}, ..., w_{in_i}\}$. Then, we use the $[CLS]$'s pooled embedding at the last layer as the feature representation of u_i.

3.3.2 GNN, RNN and DAGNN

Before introducing the DAG-ERC layers in detail, we first briefly describe graph-based models, recurrence-based models and directed acyclic graph models to help understand their differences.

For each node at each layer, graph-based models (GNN) aggregate the information of its neighboring nodes at the previous layer as follows:

$$H_i^l = f(\text{Aggregate}(\{H_j^{l-1}|j \in \mathcal{N}_i\}), H_i^{l-1}), \quad (1)$$

where $f(\cdot)$ is the information processing function, Aggregate$(\cdot)$ is the information aggregation function to gather information from neighboring nodes, and $\mathcal{N}_i$ denotes the neighbours of the i-th node.

Recurrence-based models (RNN) allow information to propagate temporally at the same layer, while the i-th node only receives information from the $(i-1)$-th node:

$$H_i^l = f(H_{i-1}^l, H_i^{l-1}). \quad (2)$$

Directed acyclic graph models (DAGNN) work like a combination of GNN and RNN. They aggregate information for each node in temporal order, and allow all nodes to gather information from neighbors and update their states at the same layer:

$$H_i^l = f(\text{Aggregate}(\{H_j^l|j \in \mathcal{N}_i\}), H_i^{l-1}). \quad (3)$$

The strength of applying DAGNN to ERC is relatively apparent: By allowing information to propagate temporally at the same layer, DAGNN can get access to distant utterances and model the information flow throughout the whole conversation, which is hardly possible for GNN. Besides, DAGNN gathers information from several neighboring utterances, which sounds more appealing than RNN as the latter only receives information from the $(i-1)$-th utterance.

3.3.3 DAG-ERC Layers

Our proposed DAG-ERC is primarily inspired by DAGNN (Thost and Chen, 2021), with novel improvements specially made for emotion recognition in conversation. At each layer l of DAG-ERC, due to the temporal information flow, the hidden state of utterances should be computed recurrently from the first utterance to the last one.

For each utterance u_i, the attention weights between u_i and its predecessors are calculated by using u_i's hidden state at the $(l-1)$-th layer to attend to the predecessors' hidden states at l-th layer:

$$\alpha_{ij}^l = \text{Softmax}_{j \in \mathcal{N}_i}(W_\alpha^l[H_j^l \| H_i^{l-1}]) \quad (4)$$

where W_α^l are trainable parameters and $\|$ denotes the concatenation operation.

The information aggregation operation in DAG-ERC is different from that in DAGNN. Instead of merely gathering information according to the attention weights, inspired by R-GCN (Schlichtkrull et al., 2018), we apply a *relation-aware feature*
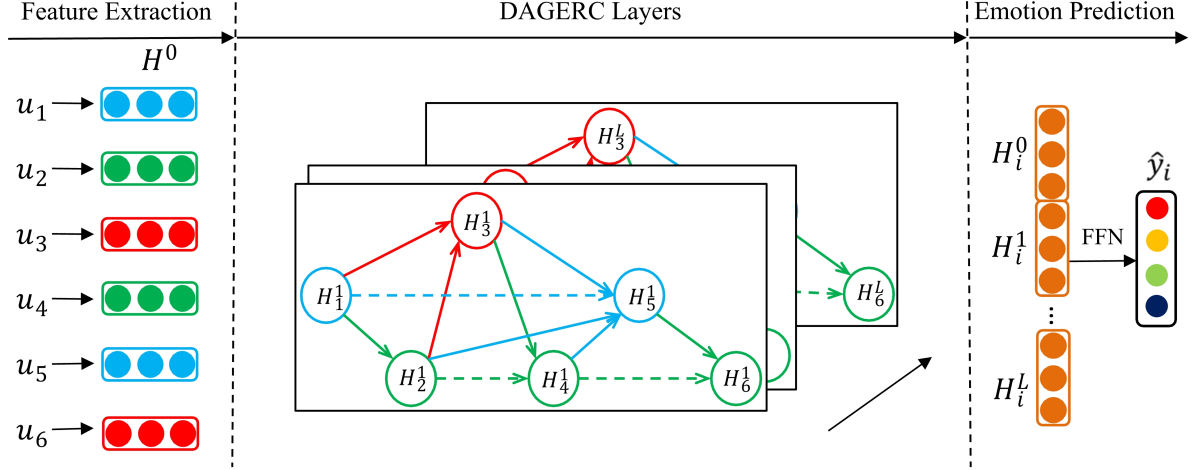
Figure 3: The framework of Directed Acyclic Graph Neural Network for ERC (DAG-ERC).

transformation to make full use of the relational type of edges:

$$M_i^l = \sum_{j \in \mathcal{N}_i} \alpha_{ij} W_{r_{ij}}^l H_j^l, \quad (5)$$

where $W_{r_{ij}}^l \in \{W_0^l, W_1^l\}$ are trainable parameters for the relation-aware transformation.

After the aggregated information M_i^l is calculated, we make it interact with u_i's hidden state at the previous layer H_i^{l-1} to obtain the final hidden state of u_i at the current layer. In DAGNN, the final hidden state is obtained by allowing M_i^l to control information propagation of H_i^{l-1} to the l-th layer with a gated recurrent unit (GRU):

$$\widetilde{H}_i^l = \mathrm{GRU}_H^l(H_i^{l-1}, M_i^l), \quad (6)$$

where H_i^{l-1}, M_i^l, and $\widetilde{H}_i^l$ are the input, hidden state and output of the GRU, respectively.

We refer to the process in Equation 6 as *nodal information unit*, because it focuses on the node information propagating from the past layer to the current layer. Nodal information unit may be suitable for the tasks that DAGNN is originally designed to solve. However, we find that only using nodal information unit is not enough for ERC, especially when the query utterance u_i's emotion should be derived from its context. The reason is that in DAGNN, the information of context M_i^l is only used to control the propagation of u_i's hidden state, and under this circumstance, the information of context is not fully leveraged. Therefore, we design another GRU called *contextual information unit* to model the information flow of historical

context through a single layer. In the contextual information unit, the roles of H_i^{i-1} and M_i^l in GRU are reversed, i.e., H_i^{i-1} controls the propagation of M_i^l:

$$C_i^l = \mathrm{GRU}_M^l(M_i^l, H_i^{l-1}). \quad (7)$$

The representation of u_i at the l-th layer is the sum of $\widetilde{H}_i^l$ and C_i^l:

$$H_i^l = \widetilde{H}_i^l + C_i^l. \quad (8)$$

3.3.4 Training and Prediction

We take the concatenation of u_i's hidden states at all DAG-ERC layers as the final representation of u_i, and pass it through a feed-forward neural network to get the predicted emotion:

$$H_i = \|_{l=0}^L H_i^l, \quad (9)$$
$$z_i = \mathrm{ReLU}(W_H H_i + b_H), \quad (10)$$
$$P_i = \mathrm{Softmax}(W_z z_i + b_z), \quad (11)$$
$$\widehat{y_i} = \mathrm{Argmax}_{k \in \mathcal{S}}(P_i[k]). \quad (12)$$

For the training of DAG-ERC, we employ the standard cross-entropy loss as objective function:

$$\mathcal{L}(\theta) = -\sum_{i=1}^M \sum_{t=1}^{N_i} \mathrm{Log} P_{i,t}[y_{i,t}], \quad (13)$$

where M is the number of training conversations, N_i is the number of utterances in the i-th conversation, $y_{i,t}$ is the ground truth label, and θ is the collection of trainable parameters of DAG-ERC.

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 four datasets.

4 Experimental Settings

4.1 Implementation Details

We conduct hyper-parameter search for our proposed DAG-ERC on each dataset by hold-out validation with a validation set. The hyper-parameters to search include learning rate, batch size, dropout rate, and the number of DAG-ERC layers. For the ω that is described in 3.2, we let $\omega = 1$ for the overall performance comparison by default, but we report the results with ω varying from 1 to 3 in 5.2. For other hyper-parameters, the sizes of all hidden vectors are equal to 300, and the feature size for the RoBERTa extractor is 1024. Each training and testing process is run on a single RTX 2080 Ti GPU. Each training process contains 60 epochs and it costs at most 50 seconds per epoch. The reported results of our implemented models are all based on the average score of 5 random runs on the test set.

4.2 Datasets

We evaluate DAG-ERC on four ERC datasets. The statistics of them are shown in Table 1.

IEMOCAP (Busso et al., 2008): A multimodal ERC dataset. Each conversation in IEMOCAP comes from the performance based on script by two actors. Models are evaluated on the samples with 6 types of emotion, namely *neutral*, *happiness*, *sadness*, *anger*, *frustrated*, and *excited*. Since this dataset has no validation set, we follow Shen et al. (2020) to use the last 20 dialogues in the training set for validation.

MELD (Poria et al., 2019b): A multimodal ERC dataset collected from the TV show *Friends*. There are 7 emotion labels including *neutral*, *happiness*, *surprise*, *sadness*, *anger*, *disgust*, and *fear*.

DailyDialog (Li et al., 2017): Human-written dialogs collected from communications of English learners. 7 emotion labels are included: *neutral*, *happiness*, *surprise*, *sadness*, *anger*, *disgust*, and *fear*. Since it has no speaker information, we consider utterance turns as speaker turns by default.

EmoryNLP (Zahiri and Choi, 2017): TV show scripts collected from *Friends*, but varies from

MELD in the choice of scenes and emotion labels. The emotion labels of this dataset include *neutral*, *sad*, *mad*, *scared*, *powerful*, *peaceful*, and *joyful*.

We utilize only the textual modality of the above datasets for the experiments. For evaluation metrics, we follow Ishiwatari et al. (2020) and Shen et al. (2020) and choose micro-averaged F1 excluding the majority class (neutral) for DailyDialog and weighted-average F1 for the other datasets.

4.3 Compared Methods

We compared our model with the following baselines in our experiments:

Recurrence-based methods: DialogueRNN (Majumder et al., 2019), DialogRNN-RoBERTa (Ghosal et al., 2020), and COSMIC without external knowledge[3] (Ghosal et al., 2020).

Graph-based methods: DialogurGCN (Ghosal et al., 2019), KET (Zhong et al., 2019), DialogXL (Shen et al., 2020) and RGAT (Ishiwatari et al., 2020).

Feature extractor: RoBERTa (Liu et al., 2019).

Previous models with our extracted features: DialogueGCN-RoBERTa, RGAT-RoBERTa and DAGNN (Thost and Chen, 2021)[4].

Ours: DAG-ERC.

5 Results and Analysis

5.1 Overall Performance

The overall results of all the compared methods on the four datasets are reported in Table 2. We can note from the results that our proposed DAG-ERC achieves competitive performances across the four datasets and reaches a new state of the art on the IEMOCAP, DailyDialog and EmoryNLP datasets.

As shown in the table, when the feature extracting method is the same, graph-based models generally outperform recurrence-based models on IEMOCAP, DailyDialog, and EmoryNLP. This phenomenon indicates that recurrence-based models cannot encode the context as effectively as graph-based models, especially for the more important local context. What's more, we see a significant improvement of DAG-ERC over the graph-based

---

[3]In this paper, we compare our DAG-ERC with COSMIC without external knowledge, rather than the complete COSMIC, in order to make a clearer comparison on the model architecture, even though our DAG-ERC outperforms the complete COSMIC on IEMOCAP, DailyDialog and EmoryNLP.

[4]DAGNN is not originally designed for ERC, so we apply our DAG building method and the extracted feature for it.

Model	IEMOCAP	MELD	DailyDialog	EmoryNLP
DialogueRNN	62.75	57.03	-	-
+RoBERTa	64.76	63.61	57.32	37.44
COSMIC	63.05	**64.28**	56.16	37.10
KET	59.56	58.18	53.37	33.95
DialogXL	65.94	62.41	54.93	34.73
DialogueGCN	64.18	58.10	-	-
+RoBERTa	64.91	63.02	57.52	38.10
RGAT	65.22	60.91	54.31	34.42
+RoBERTa	66.36	62.80	59.02	37.89
RoBERTa	63.38	62.88	58.08	37.78
DAGNN	64.61	63.12	58.36	37.89
DAG-ERC	**68.03**	63.65	**59.33**	**39.02**

Table 2: Overall performance on the four datasets.

models on IEMOCAP, which demonstrates DAG-ERC's superior ability to capture remote information given that the dialogs in IEMOCAP are much longer (almost 70 utterances per dialog).

On MELD, however, we observe that neither graph-based models nor our DAG-ERC outperforms the recurrence-based models. After going through the data, we find that due to the data collection method (collected from TV shows), sometimes two consecutive utterances in MELD are not coherent. Under this circumstance, graph-based models' advantage in encoding context is not that important.

Besides, the graph-based models see considerable improvements when implemented with the powerful feature extractor RoBERTa. In spite of this, our DAG-ERC consistently outperforms these improved graph-based models and DAGNN, confirming the superiority of the DAG structure and the effectiveness of the improvements we make to build DAG-ERC upon DAGNN.

5.2 Variants of DAG Structure

In this section, we investigate how the structure of DAG would affect our DAG-ERC's performance by applying different DAG structures to DAG-ERC. In addition to our proposed structure, we further define three kinds of DAG structure: (1) sequence, in which utterances are connected one by one; (2) DAG with single local information, in which each utterance only receives local information from its nearest neighbor, and the remote information remains the same as our DAG; (3) common DAG, in which each utterance is connected with κ previous utterances. Note that if there are only two speakers taking turns to speak in a dialog, then our DAG is equivalent to common DAG with $\kappa = 2\omega$, making the comparison less meaningful. Therefore, we conduct the experiment on EmoryNLP, where there are usually multiple speakers in one dialog, and the

DAG	# Preds	F1 score
Sequence	0.92	37.57
Single local information	1.66	38.22
Common $\kappa = 2$	1.78	38.30
Common $\kappa = 4$	3.28	38.34
Common $\kappa = 6$	4.50	38.48
Ours $\omega = 1$	2.69	**39.02**
Ours $\omega = 2$	4.46	38.90
Ours $\omega = 3$	5.65	38.94

Table 3: Different DAGs applied to DAG-ERC.

speakers speak in arbitrary order. The test performances are reported in Table 3, together with the average number of each utterance's predecessors.

Several instructive observations can be made from the experimental results. Firstly, the performance of DAG-ERC drops significantly when equipped with the sequence structure. Secondly, our proposed DAG structure has the highest performance among the DAG structures. Considering our DAG with $\omega = 2$ and common DAG with $\kappa = 6$, with very close numbers of predecessors, our DAG still outperforms the common DAG by a certain margin. This indicates that the constraints based on speaker identity and positional relation are effective inductive biases, and the structure of our DAG is more suitable for the ERC task than rigidly connecting each utterance with a fixed number of predecessors. Finally, we find that increasing the value of ω may not contribute to the performance of our DAG, and $\omega = 1$ tends to be enough.

5.3 Ablation Study

To study the impact of the modules in DAG-ERC, we evaluate DAG-ERC by removing relation-aware feature transformation, the nodal information unit, and the contextual information unit individually. The results are shown in Table 4.

As shown in the table, removing the relation-aware feature transformation causes a sharp performance drop on IEMOCAP and DailyDialog, while a slight drop on MELD and EmoryNLP. Note that there are only two speakers per dialog

Method	IEMOCAP	MELD	DailyDialog	EmoryNLP
DAG-ERC	68.03	63.65	59.33	39.02
w/o rel-trans	64.12 (↓3.91)	63.29 (↓0.36)	57.12 (↓2.21)	38.87 (↓0.15)
w/o $\widetilde{H}$	66.19 (↓1.84)	63.17 (↓0.48)	58.05 (↓1.28)	38.54 (↓0.48)
w/o C	66.32 (↓1.71)	63.36 (↓0.29)	58.90 (↓0.43)	38.50 (↓0.52)

Table 4: Results of ablation study on the four datasets, with *rel-trans*, $\widetilde{H}$, and C denoting relation-aware feature transformation, nodal information unit, and contextual information unit, respectively.
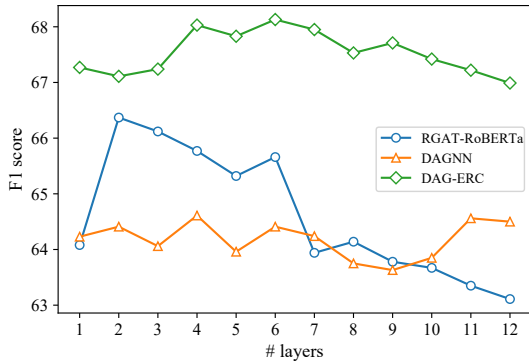
Figure 4: Test results of RGAT-RoBERTa, DAGNN, and DAG-ERC on the IEMOCAP dataset by different numbers of network layers.

Dataset	Emotional shift		w/o Emotional shift	
	# Samples	Accuracy	# Samples	Accuracy
IEMOCAP	576	57.98%	1002	74.25%
MELD	1003	59.02%	861	69.45%
DailyDialog	670	57.26%	454	59.25%
EmoryNLP	673	37.29%	361	42.10%

Table 5: Test accuracy of DAG-ERC on samples with emotional shift and without it.

in IEMOCAP and DailyDialog, and there are usually more than two speakers in dialogs of MELD and EmoryNLP. Therefore, we can infer that the relation of whether two utterances have the same speaker is sufficient for two-speaker dialogs, while falls short in the multi-speaker setting.

Moreover, we find that on each dataset, the performance drop caused by ablating nodal information unit is similar to contextual information unit, and all these drops are not that critical. This implies that either the nodal information unit or contextual information unit is effective for the ERC task, while combining the two of them can yield further performance improvement.

5.4 Number of DAG-ERC Layers

According to the model structure introduced in Section 3.3.2, the only way for GNNs to receive information from a remote utterance is to stack many GNN layers. However, it is well known that stacking too many GNN layers might cause performance degradation due to over-smoothing (Kipf and Welling, 2016). We investigate whether the same phenomenon would happen when stacking many DAG-ERC layers. We conduct an experiment on IEMOCAP and plot the test result by different numbers of layers in Figure 4, with RGAT-RoBERTa and DAGNN as baselines. As illustrated in the figure, RGAT suffers a significant performance degradation after the number of layers exceeds 6. While for DAGNN and DAG-ERC, with the number of layers changes, both of their performances fluctuate in a relatively narrow range, indicating that over-smoothing tends not to happen in the directed acyclic graph networks.

5.5 Error Study

After going through the prediction results on the four datasets, we find that our DAG-ERC fails to distinguish between similar emotions very well, such as *frustrated* vs *anger*, *happiness* vs *excited*, *scared* vs *mad*, and *joyful* vs *peaceful*. This kind of mistake is also reported by Ghosal et al. (2019). Besides, we find that DAG-ERC tends to misclassify samples of other emotions to *neutral* on MELD, DailyDialog and EmoryNLP due to the majority proportion of *neutral* samples in these datasets.

We also look closely into the *emotional shift* issue, which means the emotions of two consecutive utterances from the same speaker are different. Existing ERC models generally work poorly in emotional shift. As shown in Table 5, our DAG-ERC also fails to perform better on the samples with emotional shift than that without it, though the performance is still better than previous models. For example, the accuracy of DAG-ERC in the case of emotional shift is 57.98% on the IEMO-CAP dataset, which is higher than 52.5% achieved by DialogueRNN (Majumder et al., 2019) and 55% achieved by DialogXL (Shen et al., 2020).

6 Conclusion

In this paper, we presented a new idea of modeling conversation context with a directed acyclic graph (DAG) and proposed a directed acyclic graph neural network, namely DAG-ERC, for emotion recognition in conversation (ERC). Extensive experiments were conducted and the results show that the proposed DAG-ERC achieves comparable performance with the baselines. Moreover, by comprehensive evaluations and ablation study, we confirmed the superiority of our DAG-ERC and the impact of its modules. Several conclusions can be drawn from the empirical results. First, the DAG structures built from conversations do affect the performance of DAG-ERC, and with the constraints on speaker identity and positional relation, the proposed DAG structure outperforms its variants. Sec-

ond, the widely utilized graph relation type of whether two utterances have the same speaker is insufficient for multi-speaker conversations. Third, the directed acyclic graph network does not suffer over-smoothing as easily as GNNs when the number of layers increases. Finally, many of the errors misjudged by DAG-ERC can be accounted for by similar emotions, neutral samples and emotional shift. These reasons have been partly mentioned in previous works but have yet to be solved, which are worth further investigation in future work.

Acknowledgments

References

Miltiadis Allamanis, Earl T Barr, Premkumar Devanbu, and Charles Sutton. 2018. A survey of machine learning for big code and naturalness. *ACM Computing Surveys (CSUR)*, 51(4):1–37.

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.

Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. 2019. Semeval-2019 task 3: Emocontext contextual emotion detection in text. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 39–48.

Sheng-Yeh Chen, Chao-Chun Hsu, Chuan-Chun Kuo, Ting-Hao, Huang, and Lun-Wei Ku. 2018. Emotionlines: An emotion corpus of multi-party conversations. In *11th International Conference on Language Resources and Evaluation, LREC 2018*, pages 1597–1601.

Maxwell Crouse, Ibrahim Abdelaziz, Cristina Cornelio, Veronika Thost, Lingfei Wu, Kenneth Forbus, and Achille Fokoue. 2019. Improving graph neural network representations of logical formulae with subgraph pooling. *arXiv preprint arXiv:1911.06904*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina N. Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186.

Deepanway Ghosal, Navonil Majumder, Alexander Gelbukh, Rada Mihalcea, and Soujanya Poria. 2020. Cosmic: Commonsense knowledge for emotion identification in conversations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2470–2481.

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.

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 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2122–2132.

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.

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.

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. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 986–995.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Navonil Majumder, Pengfei Hong, Shanshan Peng, Jiankun Lu, Deepanway Ghosal, Alexander Gelbukh, Rada Mihalcea, and Soujanya Poria. 2020. Mime: Mimicking emotions for empathetic response generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8968–8979.

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.

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

Soujanya Poria, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. 2019a. Emotion recognition in conversation: Research challenges, datasets, and recent advances. *IEEE Access*, 7:100943–100953.

Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *15th International Conference on Extended Semantic Web Conference, ESWC 2018*, pages 593–607.

Björn Schuller, Michel Valster, Florian Eyben, Roddy Cowie, and Maja Pantic. 2012. Avec 2012: the continuous audio/visual emotion challenge. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 449–456.

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*.

Bing Shuai, Zhen Zuo, Bing Wang, and Gang Wang. 2016. Dag-recurrent neural networks for scene labeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3620–3629.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566.

Veronika Thost and Jie Chen. 2021. Directed acyclic graph neural networks. In *International Conference on Learning Representations*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on 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*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, pages 5753–5763.

Sayyed M. Zahiri and Jinho D. Choi. 2017. Emotion detection on tv show transcripts with sequence-based convolutional neural networks. In *AAAI Workshops*, pages 44–52.

Dong Zhang, Liangqing Wu, Changlong Sun, Shoushan Li, Qiaoming Zhu, and Guodong Zhou. 2019a. 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.

Muhan Zhang, Shali Jiang, Zhicheng Cui, Roman Garnett, and Yixin Chen. 2019b. D-vae: A variational autoencoder for directed acyclic graphs. In *Advances in Neural Information Processing Systems*, pages 1588–1600.

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.