

Relational Temporal Graph Reasoning for Dual-Task Dialogue Language Understanding

Bowen Xing and Ivor W. Tsang , *Fellow, IEEE*

Abstract—Dual-task dialog language understanding aims to tackle two correlative dialog language understanding tasks simultaneously via leveraging their inherent correlations. In this paper, we put forward a new framework, whose core is relational temporal graph reasoning. We propose a speaker-aware temporal graph (SATG) and a dual-task relational temporal graph (DRTG) to facilitate relational temporal modeling in dialog understanding and dual-task reasoning. Besides, different from previous works that only achieve implicit semantics-level interactions, we propose to model the explicit dependencies via integrating *prediction-level interactions*. To implement our framework, we first propose a novel model Dual-tAsk temporal Relational rEcurrenT Reasoning network (DARER), which first generates the context-, speaker- and temporal-sensitive utterance representations through relational temporal modeling of SATG, then conducts recurrent dual-task relational temporal graph reasoning on DRTG, in which process the estimated label distributions act as key clues in prediction-level interactions. And the relational temporal modeling in DARER is achieved by relational graph convolutional networks (RGCNs). Then we further propose Relational Temporal Transformer (ReTeFormer), which achieves fine-grained relational temporal modeling via Relation- and Structure-aware Disentangled Multi-head Attention. Accordingly, we propose DARER with ReTeFormer (DARER²), which adopts two variants of ReTeFormer to achieve the relational temporal modeling of SATG and DTRG, respectively. The extensive experiments on different scenarios verify that our models outperform state-of-the-art models by a large margin. Remarkably, on the dialog sentiment classification task in the Mastodon dataset, DARER and DARER² gain relative improvements of about 28% and 34% over the previous best model in terms of F1.

Index Terms—Dialog system, graph reasoning, language understanding, temporal relation, transformer.

I. INTRODUCTION

DIALOG language understanding [1] is the fundamental component of the dialogue system. It includes several individual tasks, e.g., dialog sentiment classification, dialog act recognition, slot filling, and (multiple) intent detection. In recent years, as researchers discover the inherent correlations among

Manuscript received 21 September 2022; revised 12 April 2023; accepted 15 June 2023. Date of publication 26 June 2023; date of current version 3 October 2023. This work was supported by Australian Research Council under Grant DP200101328. The work of Bowen Xing and Ivor W. Tsang was supported by the A*STAR Centre for Frontier AI Research. Recommended for acceptance by K. Livescu. (*Corresponding author: Bowen Xing*)

Bowen Xing and Ivor W. Tsang are with the Australian Artificial Intelligence Institute (AAII), University of Technology Sydney, Ultimo, NSW 2007, Australia, and also with CFAR, IHPC Agency for Science, Technology and Research, Singapore 138632 (e-mail: bwxing714@gmail.com; ivor.tsang@uts.edu.au).

Digital Object Identifier 10.1109/TPAMI.2023.3289509

TABLE I
A DIALOG SNIPPET FROM THE MASTODON [3] DATASET

Utterances	Act	Sentiment
u_a : I highly recommend it. Really awesome progression and added difficulty	Statement	Positive
u_b : I never have.	Disagreement	Negative

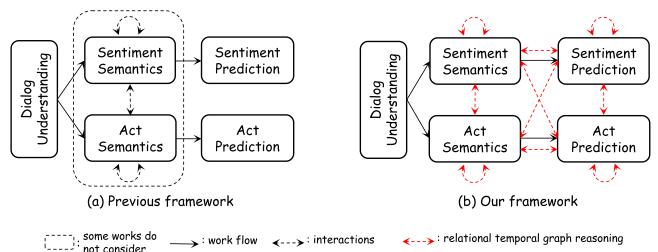


Fig. 1. Illustration of previous framework and ours.

some specific task-pair, the joint task which tackles two tasks simultaneously has attracted increasing attention. For example, dialog sentiment classification (DSC) and dialog act recognition (DAR) are two challenging tasks in dialog systems [2], while the task of joint DSC and DAR aims to simultaneously predict the sentiment label and act label for each utterance in a dialog [3], [4]. An example is shown in Table I. To predict the sentiment of u_b , besides its *semantics*, its *Disagreement act label* and the *Positive sentiment label* of its *previous* utterance (u_a) can provide useful references, which contribute a lot when humans do this task. This is because the *Disagreement act label* of u_b denotes it has the opposite opinion with u_a , and thus u_b tends to have a *Negative sentiment label*, the opposite one with u_a (*Positive*). Similarly, the opposite sentiment labels of u_b and u_a are helpful to infer the *Disagreement act label* of u_b . In this paper, we term this process as dual-task reasoning, where there are three key factors: 1) the semantics of u_a and u_b ; 2) the temporal relation between u_a and u_b ; 3) u_a 's and u_b 's labels for another task.

In previous works, different models are proposed to model the correlations between DSC and DAR. [3] propose a multi-task model in which the two tasks share a single encoder. [4], [5], [6], [7] try to model the semantics-level interactions of the two tasks. The framework of previous models is shown in Fig. 1(a). For dialog understanding, Co-GAT [7] applies graph attention network (GAT) [8] over an undirected disconnected graph which consists of isolated speaker-specific full-connected subgraphs.

Therefore, it suffers from the issue that the inter-speaker interactions cannot be modeled, and the temporal relations between utterances are omitted. For dual-task reasoning, on the one hand, previous works only consider the parameter sharing and semantics-level interactions, while the label information is not explicitly integrated into the dual-task interactions. Consequently, the explicit dependencies between the two tasks cannot be captured and previous dual-task reasoning processes are inconsistent with human intuition, which leverages the label information as crucial clues. On the other hand, previous works do not consider the temporal relations between utterances in dual-task reasoning, in which they play a key role.

In this paper, we try to address the above issues by introducing temporal relations and leveraging label information. To introduce temporal relations, we design a speaker-aware temporal graph (SATG) for dialog understanding, and a dual-task reasoning temporal graph (DRTG) for dual-task relational reasoning. Intuitively, different speakers' semantic states will change as the dialog goes, and these semantic state transitions trigger different sentiments and acts. SATG is designed to model the speaker-aware semantic states transitions, which provide essential indicative semantics for both tasks. In SATG, there is one group of utterance nodes and two kinds of temporal relations: previous and future. Since the temporal relation is a key factor in dual-task reasoning, DRTG is designed to integrate inner- and inter-task temporal relations, making the dual-task reasoning process more rational and effective. In SATG, there are two parallel groups of utterance nodes and three kinds of temporal relations: previous, future, and equal.

To leverage label information, we propose a new framework, as shown in Fig. 1(b). Except for semantics-level interactions, it integrates several kinds of prediction-level interactions. First, self-interactions of sentiment predictions and act predictions. In both tasks, there are prediction-level correlations among the utterances in a dialog. In the DSC task, the sentiment state of each speaker tends to be stable until the utterances from others trigger the changes [9]. In the DAR task, there are different patterns (e.g., Questions-Inform and Directives-Commissives) reflecting the interactions between act labels [10]. Second, interactions between the predictions and semantics. Intuitively, the predictions can offer feedback to semantics, which can rethink and then reversely help revise the predictions. Third, prediction-prediction interactions between DSC and DAR, which model the explicit dependencies. However, since our objective is to predict the labels of both tasks, there is no ground-truth label available for prediction-level interactions. To this end, we design a recurrent dual-task reasoning mechanism that leverages the label distributions estimated in the previous step as prediction clues of the current step for producing new predictions. In this way, the label distributions of both tasks are gradually improved along the step. To implement our framework, we propose Dual-tAsk temporal Relational rEcurrent Reasoning Network¹ (DARER) [11], which includes three main components. The *Dialog Understanding* module conducts relation-specific

graph transformations (RSGT) over SATG to generate context-, speaker- and temporal-sensitive utterance representations. The *Initial Estimation* module produces the initial label information which is fed to the *Recurrent Dual-task Reasoning* module, in which RSGT operates on DRTG to conduct dual-task relational reasoning. And the RSGTs are achieved by relational graph convolutional networks [12]. Moreover, we design logic-heuristic training objectives to force DSC and DAR to gradually prompt each other in the recurrent dual-task reasoning process.

Then we further propose Relational Temporal Transformer (ReTeFormer) and DARER². The main difference between DARER and DARER² is that in DARER² the original RGCNs applied over SATG and DRTG are replaced with our proposed SAT-ReTeFormer and DTR-ReTeFormer. The core of ReTeFormer is the Relation- and Structure-Aware Disentangled Multi-head Attention, which can achieve fine-grained relational temporal modeling. Generally, DARER² has three distinguished advantages over DARER: 1) ReTeFormer integrates dialog structural information, achieving more comprehensive and fine-grained relational temporal graph reasoning; 2) the relational temporal attention mechanism of ReTeFormer can comprehensively and explicitly model the correlations among dual tasks semantics and predictions; 3) the relation specific attention maps derived by ReTeFormer can provide explainable evidence of relational temporal graph reasoning, making the model more reliable.

In summary, this work has three major contributions. 1) We propose DARER, which is based on a new framework that for the first time achieves relational temporal graph reasoning and prediction-level interactions. 2) Our proposed DARER² further improves the relational temporal graph reasoning with our proposed ReTeFormer which is based on the Relation- and Structure-Aware Disentangled Multi-head Attention. 3) Experiments prove that DARER and DARER² significantly outperform the state-of-the-art models in different scenarios of dual-task dialog language understanding.

The remainder of this paper is organized as follows. In Section II, the related work of two scenarios of dual-task dialog language understanding, 1) Joint Dialog Sentiment Classification and Act Recognition 2) Joint Multiple Intent Detection and Slot Filling, are summarized, and the differences of our method from previous studies are highlighted. Section III elaborates on the overall model architecture shared by DARER and DARER². Sections IV and V describe DARER and DARER², respectively. Experimental results are reported and analyzed in Section VI, and note that the task definition of Joint Multiple Intent Detection and Slot Filling as well as the experiments on this task are introduced in Section VI-J. Finally, we conclude this work and provide some prospective future directions in Section VII.

II. RELATED WORKS

In recent years, researchers have discovered that some dialog language understanding tasks are correlative and they can be tackled simultaneously by leveraging their beneficial correlations.

¹The content of DARER was presented as a poster in ACL 2022 conference.

A. Joint Dialog Sentiment Classification and Act Recognition

Dialog Sentiment Classification [13], [14], [15], [16], [17] and Dialog Act Recognition [18], [19], [20] are both utterance-level classification tasks. Recently, it has been found that these two tasks are correlative, and they can work together to indicate the speaker's more comprehensive intentions [5]. With the development of well-annotated corpora [3], [10], in which both the act label and sentiment label of each utterance are provided, several models have been proposed to tackle the joint dialog sentiment classification and act recognition task. Cerisara et al. [3] propose a multi-task framework based on a shared encoder that implicitly models the dual-task correlations. Kim and Kim [5] integrate the identifications of dialog acts, predictors and sentiments into a unified model. To explicitly model the mutual interactions between the two tasks, Qin et al. [4] propose a stacked co-interactive relation layer, and Li et al. [6] propose a context-aware dynamic convolution network to capture the crucial local context. More recently, Qin et al. [7] propose Co-GAT, which applies graph attention on a fully-connected undirected graph consisting of two groups of nodes corresponding to the two tasks, respectively.

This work is different from previous works on three aspects. First, we model the inner- and inter-speaker temporal dependencies for dialog understanding. Second, we model the cross- and self-task temporal dependencies for dual-task reasoning; Third, we achieve prediction-level interactions in which the estimated label distributions act as important and explicit clues other than semantics.

B. Joint Multiple Intent Detection and Slot Filling

It has been widely recognized that intent detection and slot filling have strong correlations. And a group of models [21], [22], [23], [24], [25], [26], [27], [28], [29], [30] have been proposed to leverage the correlations for tackling the joint task of intent detection and slot filling in the multi-task framework. However, in real-word scenarios, a single utterance usually expresses multiple intents, which cannot be handled by the above models. To this end, Kim et al. [31] propose a multi-intent spoken language understanding model. Besides, Gangadharaiyah and Narayanaswamy [32] propose the first model that utilizes a slot-gate mechanism to jointly tackle the tasks of multiple intent detection and slot filling. Furthermore, Qin et al. [33] propose an adaptive graph-interactive model to model the fine-grained multiple intent information and integrate it into slot filling task via GAT. More recently, Qin et al. [34] propose to conduct non-autoregressive slot decoding in a parallel manner for slot filling, and the proposed GL-GIN achieves state-of-the-art performance.

Our work can be generalized to the joint task of multiple intent detection and slot filling. Existing methods adopt homogeneous graphs and vanilla GATs to achieve the interactions between the predicted intents and slot semantics, ignoring the specific relations among the two tasks' nodes and the temporal dependencies among the slot nodes. Different from them, our method achieves relational temporal graph reasoning.

III. OVERALL MODEL ARCHITECTURE

Given a dialog consisting of N utterances: $\mathcal{D} = (u_1, u_2, \dots, u_N)$, our objective is to predict both the dialog sentiment labels $Y^S = y_1^s, \dots, y_N^s$ and the dialog act labels $Y^A = y_1^a, \dots, y_N^a$ in a single run.

The overall network architecture shared by DARER and DARER² is shown in Fig. 2. It consists of three modules, whose details are introduced in this section.

A. Dialog Understanding

1) *Utterance Encoding*: In previous works, BiLSTM [35] is widely adopted as the utterance encoder to generate the initial utterance representation: $H = (h_0, \dots, h_N)$. In this paper, besides BiLSTM, we also study the effect of different pre-trained language model (PTLM) encoders in Section VI-C3.

BiLSTM: We apply the BiLSTM over the word embeddings of u_t to capture the inner-sentence dependencies and temporal relationships among the words, producing a series of hidden states $H_{u,i} = (h_{u,i}^0, \dots, h_{u,i}^{l_i})$, where l_i is the length of u_i . Then we feed $H_{u,i}$ into a max-pooling layer to get the representation for each u_i .

PTLM: We separately feed each utterance into the PTLM encoder and take the output hidden state of the [CLS] token as the utterance representation.

2) *Speaker-Aware Temporal RSGT*: To capture the inter- and intra-speaker semantic interactions and the speaker-aware temporal dependencies between utterances, we conduct Speaker-aware Temporal relation-specific graph transformations (SAT-RSGT). Now we obtain the context-, speaker- and temporal-sensitive utterance representations: $\hat{H} = (\hat{h}_0, \dots, \hat{h}_N)$.

B. Initial Estimation

To obtain task-specific utterances representations, we separately apply two BiLSTMs over \hat{H} to obtain the utterance hidden states for sentiments and acts respectively: $H_s^0 = \text{BiLSTM}_S(\hat{H})$, $H_a^0 = \text{BiLSTM}_A(\hat{H})$, where $H_s^0 = \{h_{s,i}^0\}_{i=1}^N$ and $H_a^0 = \{h_{a,i}^0\}_{i=1}^N$. Then H_s^0 and H_a^0 are separately fed into Sentiment Decoder and Act Decoder to produce the initial estimated label distributions:

$$\begin{aligned} P_S^0 &= \{P_{S,i}^0\}_{i=1}^N \quad P_A^0 = \{P_{A,i}^0\}_{i=1}^N \\ P_{S,i}^0 &= \text{softmax}(W_d^s h_{s,i}^0 + b_d^s) \\ &= [p_{s,i}^0[0], \dots, p_{s,i}^0[k], \dots, p_{s,i}^0(|\mathcal{C}_s| - 1)] \\ P_{A,i}^0 &= \text{softmax}(W_d^a h_{a,i}^0 + b_d^a) \\ &= [p_{a,i}^0[0], \dots, p_{a,i}^0[k], \dots, p_{a,i}^0(|\mathcal{C}_a| - 1)] \end{aligned} \quad (1)$$

where W_d^* and b_d^* are weight matrices and biases, \mathcal{C}_s and \mathcal{C}_a are sentiment class set and act class set.

C. Recurrent Dual-Task Reasoning

At step t , the recurrent dual-task reasoning module takes two streams of inputs: 1) hidden states $H_s^{t-1} \in \mathbb{R}^{N \times d}$ and $H_a^{t-1} \in \mathbb{R}^{N \times d}$; 2) label distributions $P_S^{t-1} \in \mathbb{R}^{N \times |\mathcal{C}_s|}$ and $P_A^{t-1} \in \mathbb{R}^{N \times |\mathcal{C}_a|}$.

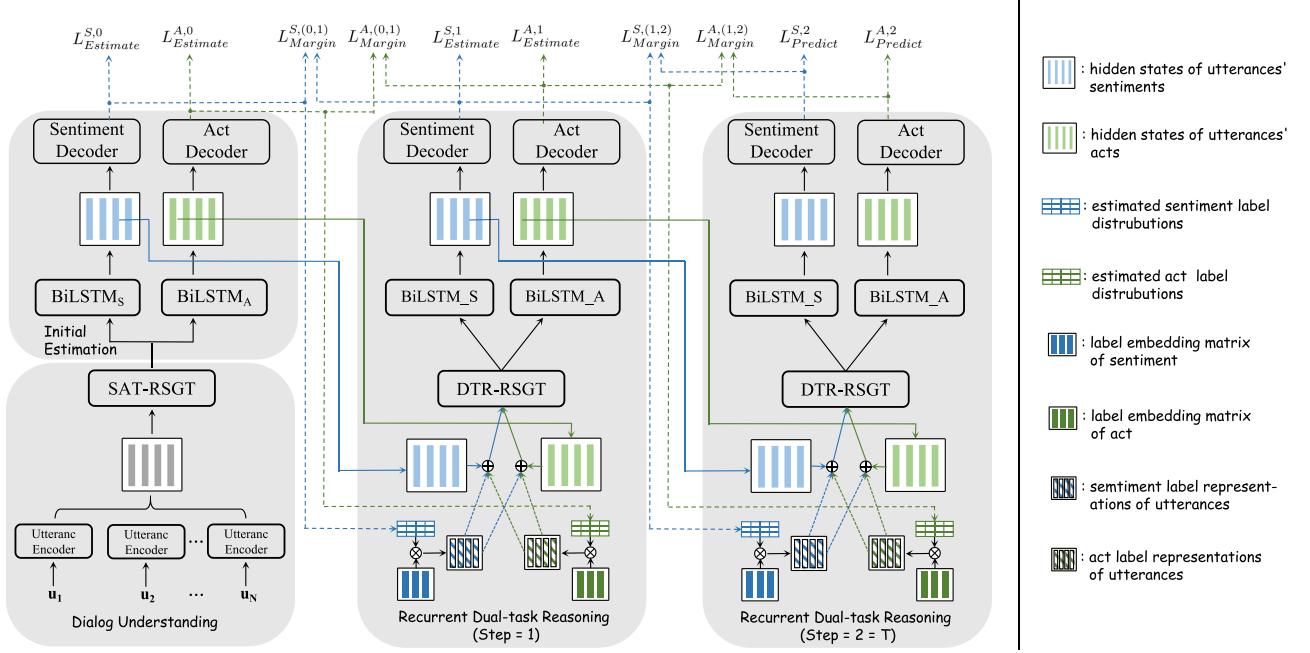


Fig. 2. Overall network architecture of DARER and DARER². In DARER, SAT-RSGT and DTR-RSGT are achieved by RGCNs, while in DARER², they are achieved by SAT-ReTeFormer and DTR-ReTeFormer, respectively. Without loss of generality, the step number T in this illustration is set 2.

1) *Projection of Label Distribution*: To achieve the prediction-level interactions, we should represent the label information in vector form to let it participate in calculations. We use P_S^{t-1} and P_A^{t-1} to respectively multiply the sentiment label embedding matrix $M_s^e \in \mathbb{R}^{|\mathcal{C}_s| \times d}$ and the act label embedding matrix $M_a^e \in \mathbb{R}^{|\mathcal{C}_a| \times d}$, obtaining the sentiment label representations $E_S^t = \{e_{s,i}^t\}_{i=1}^N$ and act label representations $E_A^t = \{e_{a,i}^t\}_{i=1}^N$. In particular, for each utterance, its sentiment label representation and act label representation are computed as:

$$\begin{aligned} e_{s,i}^t &= \sum_{k=0}^{|\mathcal{C}_s|-1} p_{s,i}^{t-1}[k] \cdot v_s^k \\ e_{a,i}^t &= \sum_{k'=0}^{|\mathcal{C}_a|-1} p_{a,i}^{t-1}[k'] \cdot v_a^{k'} \end{aligned} \quad (2)$$

where v_s^k and $v_a^{k'}$ are the label embeddings of sentiment class k and act class k' , respectively.

2) *Dual-Task Reasoning RSGT*: To achieve the self- and mutual-interactions between the semantics and predictions, for each node in DRTG, we superimpose its corresponding utterance's label representations of both tasks on its hidden state:

$$\begin{aligned} \hat{h}_{s,i}^t &= h_{s,i}^{t-1} + e_{s,i}^t + e_{a,i}^t \\ \hat{h}_{a,i}^t &= h_{a,i}^{t-1} + e_{s,i}^t + e_{a,i}^t \end{aligned} \quad (3)$$

Thus the representation of each node contains the task-specific semantic features and both tasks' label information, which are then incorporated into the relational reasoning process to achieve semantics-level and prediction-level interactions.

The obtained \hat{H}_s^t and \hat{H}_a^t both have N vectors, respectively corresponding to the N sentiment nodes and N act nodes on DRTG. Then we feed them into the Dual-task Reasoning relation-specific graph transformations (DTR-RSGT) conducted on DRTG. Now we get \bar{H}_s^t and \bar{H}_a^t .

3) *Label Decoding*: For each task, we use a task-specific BiLSTM (TS-LSTM) to generate a new series of task-specific hidden states:

$$\begin{aligned} H_s^t &= \text{BiLSTM}_S(\bar{H}_s^t) \\ H_a^t &= \text{BiLSTM}_A(\bar{H}_a^t) \end{aligned} \quad (4)$$

Besides, as \bar{H}_s^t and \bar{H}_a^t both contain the label information of the two tasks, the two TS-LSTMs have another advantage of label-aware sequence reasoning, which has been proven can be achieved by LSTM [36].

Then H_S^t and H_A^t are separately fed to Sentiment Decoder and Act Decoder to produce P_S^t and P_A^t .

D. Training Objective

Intuitively, there are two important logic rules in our model. First, the produced label distributions should be good enough to provide useful label information for the next step. Otherwise, noisy label information would be introduced, misleading the dual-task reasoning. Second, both tasks are supposed to learn more and more beneficial knowledge from each other in the recurrent dual-task reasoning process. Scilicet the estimated label distributions should be gradually improved along steps. In order to force our model to obey these two rules, we propose a constraint loss $L_{Constraint}$ that includes two terms: $L_{Estimate}$ and L_{Margin} , which correspond to the two rules, respectively.

Estimate Loss $L_{Estimate}$ is the cross-entropy loss forcing model to provide good enough label distributions for the next step. At step t , for DSC task, $\mathcal{L}_{Estimate}^{S,t}$ is defined as:

$$\mathcal{L}_{Estimate}^{S,t} = \sum_{i=1}^N \sum_{k=0}^{|\mathcal{C}_s|-1} y_i^s[k] \log(p_{s,i}^t[k]) \quad (5)$$

Margin Loss L_{Margin} works on the label distributions of two adjacent steps, and it promotes the two tasks gradually learning beneficial knowledge from each other via forcing DARER to produce better predictions at step t than step $t-1$. Besides, although the model can receive more information at step t than $t-1$, this information is imperfect because there are some incorrect predictions of the previous step. Therefore, we use the margin loss to force the model to leverage the *beneficial information* to output better predictions. For DSC task, $\mathcal{L}_{Margin}^{S,(t,t-1)}$ is a margin loss defined as:

$$\mathcal{L}_{Margin}^{S,(t,t-1)} = \sum_{i=1}^N \sum_{k=0}^{|\mathcal{C}_s|-1} y_i^s[k] \max(0, p_{s,i}^{t-1}[k] - p_{s,i}^t[k]) \quad (6)$$

If the correct class's possibility at step t is worse than at step $t-1$, $\mathcal{L}_{Margin}^{S,(t,t-1)} > 0$. Then the negative gradient further force the model to predict better at step t . Otherwise, the correct class's possibility at step t is better than or equal to the one at step $t-1$. In this case, $\mathcal{L}_{Margin}^{S,(t,t-1)} = 0$.

Constraint loss $L_{Constraint}$ is the weighted sum of $L_{Estimate}$ and L_{Margin} , with a hyper-parameter γ_s balancing the two kinds of punishments. For DSC task, $\mathcal{L}_{Constraint}^S$ is defined as:

$$\mathcal{L}_{Constraint}^S = \sum_{t=0}^{T-1} \mathcal{L}_{Estimate}^{S,t} + \gamma_s * \sum_{t=1}^T \mathcal{L}_{Margin}^{S,(t,t-1)} \quad (7)$$

Final Training Objective The total loss for DSC task (\mathcal{L}^S) is the sum of $\mathcal{L}_{Constraint}^S$ and $\mathcal{L}_{Prediction}^S$:

$$\mathcal{L}^S = \mathcal{L}_{Prediction}^S + \mathcal{L}_{Constraint}^S \quad (8)$$

where $\mathcal{L}_{Prediction}^S$ is the cross-entropy loss of the produced label distributions at the final step T :

$$\mathcal{L}_{Prediction}^S = \sum_{i=1}^N \sum_{k=0}^{|\mathcal{C}_s|-1} y_{s,i} \log(p_{s,i}^T[k]) \quad (9)$$

The total loss of DAR task (\mathcal{L}^A) can be derived similarly like (5) to (9).

The final training objective of our model is the sum of the total losses of the two tasks:

$$\mathcal{L} = \mathcal{L}^S + \mathcal{L}^A \quad (10)$$

IV. DARER

Based on the overall model introduced in Section III, DARER achieves SAT-RSGT and DTR-RSGT via applying RGCNs on the speaker-aware temporal graph (SATG) and dual-task reasoning temporal graph (DRTG), respectively.

TABLE II
ALL RELATION TYPES IN SATG (ASSUME THERE ARE TWO SPEAKERS)

r_{ij}	1	2	3	4	5	6	7	8
$I_s(i)$	1	1	1	1	2	2	2	2
$I_s(j)$	1	1	2	2	1	1	2	2
$pos(i,j)$	>	\leq	>	\leq	>	\leq	>	\leq

$I_s(i)$ indicates the speaker node i is from. $pos(i,j)$ indicates the relative position of node i and j .

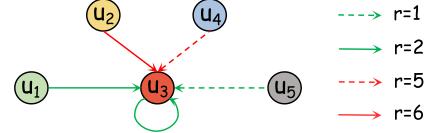


Fig. 3. Example of SATG. u_1, u_3 and u_5 are from speaker 1 while u_2 and u_4 are from speaker 2. w.l.o.g, only the edges directed into u_3 node are illustrated.

A. SAT-RSGT

1) *Speaker-Aware Temporal Graph*: We design a SATG to model the information aggregation between utterances in a dialog. Formally, SATG is a complete directed graph denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$. In this paper, the nodes in \mathcal{G} are the utterances in the dialog, i.e., $|\mathcal{V}| = N, \mathcal{V} = (u_1, \dots, u_N)$, and the edge $(i, j, r_{ij}) \in \mathcal{E}$ denotes the information aggregation from u_i to u_j under the relation $r_{ij} \in \mathcal{R}$. Table II lists the definitions of all relation types in \mathcal{R} . In particular, there are three kinds of information conveyed by r_{ij} : the speaker of u_i , the speaker of u_j , and the relative position of u_i and u_j . Naturally, the utterances in a dialog are chronologically ordered, so the relative position of two utterances denotes their temporal relation. An example of SATG is shown in Fig. 3. Compared with the previous dialog graph structure [4], [7], our SATG has two main advancements. First, as a complete directed graph, SATG can model both the intra- and inter-speaker semantic interactions. Second, incorporating temporal information, SATG can model the transitions of speaker-aware semantic states as the dialog goes on, which benefits both tasks.

2) *SAT-RGCN*: Inspired from [12], we apply SAT-RGCN over SATG to achieve the information aggregation:

$$\hat{h}_i = W_1 h_i^0 + \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}_i^r} \frac{1}{|\mathcal{N}_i^r|} W_1^r h_j^0 \quad (11)$$

where W_1 is self-transformation matrix and W_1^r is relation-specific matrix.

B. DTR-RSGT

1) *Dual-Task Reasoning Temporal Graph*: Inspired by [37], [38], [39], [40], we design a DRTG, as shown in Fig. 4, to provide an advanced platform for dual-task relational reasoning. It is also a complete directed graph that consists of $2N$ dual nodes: N sentiment nodes and N act nodes. The definitions of all relation types in \mathcal{R}' are listed in Table III. Intuitively, when predicting the label of a node, the information of its dual node plays a key role, so we emphasize the temporal relation of ' $=$ ' rather than merge it with ' $<$ ' like SATG. Specifically,

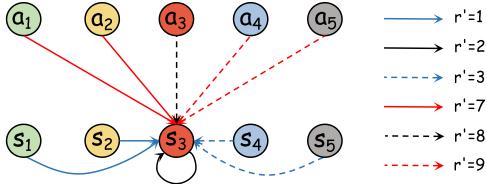


Fig. 4. Example of DRTG. s_i and a_i respectively denote the node of DAC task and DAR task. w.l.o.g, only the edges directed into s_3 are illustrated.

TABLE III
ALL RELATION TYPES IN DRTG

r'_{ij}	1	2	3	4	5	6	7	8	9	10	11	12
$I_t(i)$	S	S	S	S	S	S	A	A	A	A	A	A
$I_t(j)$	S	S	S	A	A	A	S	S	S	A	A	A
$pos(i, j)$	<	=	>	<	=	>	<	=	>	<	=	>

$I_t(i)$ indicates that node i is a sentiment (S) node or act (A) node.

the relation r'_{ij} conveys three kinds of information: the task of n_i , the task of n_j and the temporal relation between n_i and n_j . An example of DRTG is shown in Fig. 3. Compared with the previous dual-task graph structure [4], [7], our DRTG has two major advancements. First, the temporal relations in DRTG can make the DTR-RSGT capture the temporal information, which is essential for dual-task reasoning, while this cannot be achieved by the co-attention [4] or graph attention network [7] operating on their non-temporal graphs. Second, in DRTG, the information aggregated into a node is decomposed by different relations that correspond to individual contributions, rather than only depending on the semantic similarity measured by the attention mechanisms.

2) *DTR-RSGT*: We apply DTR-RGCN to DRTG to achieve information aggregation. Specifically, the node updating process of DTR-RGCN can be formulated as:

$$\bar{h}_i^t = W_2 \hat{h}_i^t + \sum_{r \in \mathcal{R}'} \sum_{j \in \mathcal{N}_i^{r'}} \frac{1}{|N_i^{r'}|} W_2^r \hat{h}_j^t \quad (12)$$

where W_2 is self-transformation matrix and W_2^r is relation-specific matrix.

V. DARER²

Based on the overall model introduced in Section III, DARER² achieves SAT-RSGT and DTR-RSGT via applying our proposed ReTeFormers on the speaker-aware temporal graph (SATG) and dual-task reasoning temporal graph (DRTG), respectively. Next, we first introduce the details of ReTeFormer, then the SAT-RSGT and DTR-RSGT of DARER².

A. Relational Temporal Transformer

The architecture of ReTeFormer is shown in Fig. 5. The core of ReTeFormer is the Relation- and Structure- Aware Disentangled Multi-head Attention, which can handle the Relation Modeling and Temporal Modeling simultaneously.

Relational Modeling: The relational graph can be disentangled into different views, each of which corresponds to a specific relation and has its own adjacency matrix. In ReTeFormer, each

head of Relational Temporal Attention corresponds to a specific relation and has its own parameterization, so as to achieve the relation-specific information aggregation. To make sure that the information aggregation of each relation is along the relation-specific structure, we design the Relation- and Structure-Aware 2-D Mask which uses the relation-specific adjacency matrix to mask the correlation matrix. Since the final node representation receives information along multiple relations, we design the Dynamic 1-D Mask and Merge module to extract and sum each node's sub-representations obtained from multi-head Relational Temporal Attentions.

Temporal Modeling: A dialog can be regarded as a temporal sequence of utterances. ReTeFormer utilizes position embedding to achieve temporal modeling. The position embedding is one of the foundations of Transformer [41], which adds the position embedding to the input representation. However, recently it has been proven that adding together the position embeddings and word embeddings at input harms the attention and further limit the model's expressiveness because this operation brings mixed correlations between the two heterogeneous information resources (semantics and position) and unnecessary randomness in the attention [42]. To this end, Ke et al. (2021) [42] propose to model word contextual correlation and positional correlation separately with different parameterizations and then add them together. And our ReTeFormer follows this manner.

Next, we introduce the details of Relation- and Structure-Aware Disentangled Multi-head Attention, which is the core of our ReTeFormer.²

1) *Relation-Specific Scaled Dot-Product Attention*: For the head corresponding to relation r , the correlation score $\hat{\alpha}_{ij}^r$ between every two nodes is obtained via relational modeling and temporal modeling:

$$\begin{aligned} \hat{\alpha}_{ij}^r &= \frac{1}{\sqrt{d}} (Q_{[h,i]}^r) (K_{[h,j]}^r)^T + \frac{1}{\sqrt{d}} (Q_{[p,i]}^r) (K_{[p,j]}^r)^T \\ Q_{[h,i]}^r &= h_i W_Q^r, \quad K_{[h,j]}^r = h_j W_K^r \\ Q_{[p,i]}^r &= p_i U_Q^r, \quad K_{[p,j]}^r = p_j U_K^r \end{aligned} \quad (13)$$

where h_* p_* denote the input hidden state and position embedding, respectively; W_Q^r and W_K^r denote the relation-specific projection matrix for the hidden states; U_Q^r and U_K^r denote the relation-specific projection matrix for the position embeddings; \sqrt{d} is the scaling term for retaining the magnitude of $\hat{\alpha}_{ij}^r$.

Now we obtain the relation-specific correlation matrix M^r for each relation r , which represents each two nodes' correlation along the specific relation.

2) *Relation- and Structure-Aware 2-D Mask*: Although we obtain the relation-specific correlation scores of all node pairs, a specific relation has its own adjacent structure which is crucial for information aggregation. And the attention score between two nodes should also be calculated regarding the relation-specific neighbors. To achieve this, we design the relation- and structure-aware 2-D mask to introduce the relation-specific

²In this section, we omit the introduction of the residual connection, the layer normalization, and the feed-forward layers, whose details are the same as vanilla Transformer [41].

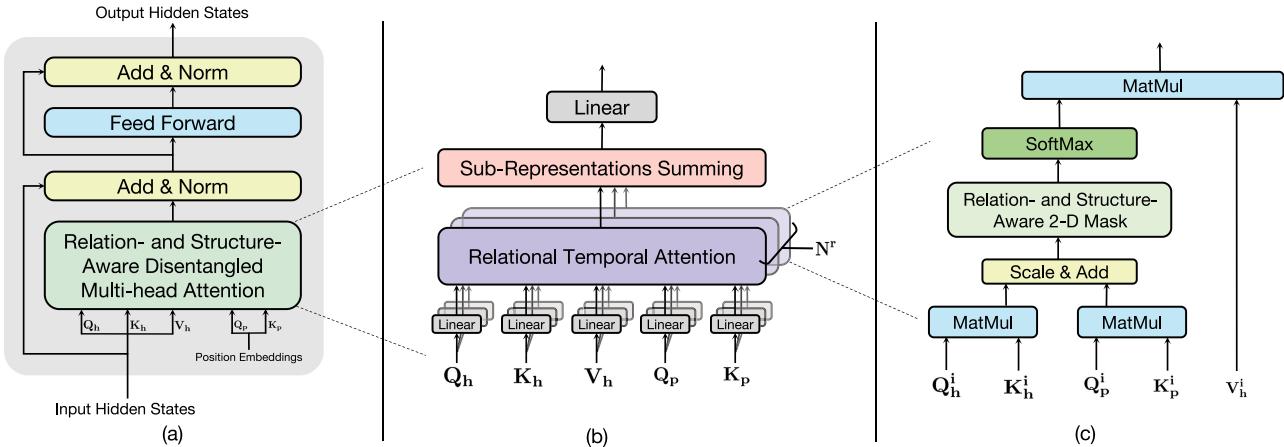


Fig. 5. (a) Illustration of ReTeFormer. (b) Illustration of Relation- and Structure-Aware Disentangled Multi-head Attention. (c) Illustration of Relational Temporal Attention corresponding to the i th relation. \mathbf{Q}_h , \mathbf{K}_h and \mathbf{V}_h denote the query matrix, key matrix and value matrix of input hidden states. \mathbf{Q}_p and \mathbf{K}_p denote the query matrix and key matrix of the absolute position embeddings. \mathbf{N}^r denotes the number of relations.

structure into the attention mechanism. Specifically, the relational graph can derive N^r relation-specific adjacency matrix via disentangling. And for each relation r , its adjacency matrix A^r is used to mask its correlation matrix M^r . Finally, the normalized relation-specific attention score α_{ij}^r is obtained as follows:

$$\alpha_{ij}^r = \text{softmax}(f_{mask}^{2D}(\alpha_{ij}^r, A_{ij}^r))$$

$$f_{mask}^{2D} = \begin{cases} \hat{\alpha}_{ij}^r & A_{ij}^r = 1 \\ -\infty & A_{ij}^r = 0 \end{cases} \quad (14)$$

where f_{mask}^{2D} denotes the function of the relation- and structure-aware 2-D mask.

3) *Output Node Representation*: For the attention head corresponding to relation r , the updated sub-representation of node i (or the information that node i should receive along relation r) is:

$$\hat{h}_i^r = \sum_{j \in \mathcal{N}_i^r} \alpha_{ij}^r V_{[h,j]}^r$$

$$V_{[h,j]}^r = h_j W_V^r \quad (15)$$

A node is always connected to other nodes along different relations. Therefore, the final updated representation node i is the sum of its sub-representations of all attention heads:

$$\hat{h}_i = \sum_{r \in \mathcal{R}} \hat{h}_i^r \quad (16)$$

B. Sat-Reteformer

In DARER², the speaker-aware temporal RSGT (Section III-A2) is achieved by the SAT-ReTeFormer rather than the RGCN used in DARER. And the speaker-aware graph for SAT-ReTeFormer is shown in Fig. 6. The input of SAT-ReTeFormer is the sequence of the initial utterance representations $H = (h_0, \dots, h_N)$. In the SAT-ReTeFormer's speaker-aware graph, each node corresponds to an utterance, whose representation h_i corresponds to its position embedding p_i . Compared with RGCN, our SAT-ReTeFormer can explicitly

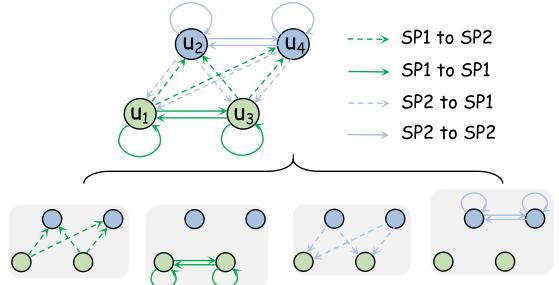


Fig. 6. Example of the speaker-aware graph for SAT-ReTeFormer and its four disentangled views. Assuming there are four utterances: u_1 and u_3 (in green color) are from the speaker 1 (SP1); u_2 and u_4 (in blue color) are from the speaker 2 (SP2). Each view has its own adjacency matrix for the corresponding head of relational temporal attention.

model the correlations among the utterances, integrating both the speaker information and the fine-grained temporal information. After SAT-ReTeFormer, we obtain the context-, speaker- and temporal-sensitive utterance representations: $\hat{H} = (\hat{h}_0, \dots, \hat{h}_N)$.

C. Dtr-Reteformer

In DARER², the dual-task reasoning RSGT (Section III-C2) is achieved by DTR-ReTeFormer rather than the RGCN used in DARER. And the dual-task reasoning graph for DTR-ReTeFormer is illustrated in Fig 7.

The input of DTR-ReTeFormer is the concatenation of \hat{H}_s^t and \hat{H}_a^t : $\hat{H}_s^t \| \hat{H}_a^t = [\hat{h}_{s,1}^t, \dots, \hat{h}_{s,N}^t, \hat{h}_{a,1}^t, \dots, \hat{h}_{a,N}^t]$. Since $\hat{h}_{s,i}^t$ and $\hat{h}_{a,i}^t$ corresponds to the same utterance (u_i), they have the same position embedding p_i .

Here we demonstrate the details of the semantics- and prediction-level interactions achieved by DTR-ReTeFormer. Assuming that node i is sentiment node and node j is act node, the correlative score between i and j is calculated as follows

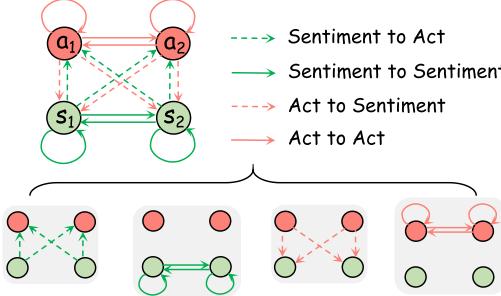


Fig. 7. Example of the dual-task reasoning graph for SAT-ReTeFormer. W.l.o.g, the dialog includes two utterances. s_1 and s_2 (in green color) denote the sentiment nodes corresponding to the first and second utterances; a_1 and a_2 (in red color) denote the act nodes of the first and second utterances. The graph can be disentangled into four views along the four relations, and each view has its own adjacency matrix used in the corresponding head of relational temporal attention.

(bringing (3) into (13)):

$$\begin{aligned}
 \hat{\alpha}_{ij}^{r',t} &= \frac{1}{\sqrt{d}}(\hat{h}_{s,i}^t W_Q^r)(\hat{h}_{a,j}^t W_K^r)^T + \frac{1}{\sqrt{d}}(p_j U_Q^r)(p_j U_K^r)^T \\
 &= \frac{1}{\sqrt{d}}((h_{s,i}^{t-1} + e_{s,i}^t + e_{a,i}^t)W_Q^r)((h_{a,j}^{t-1} + e_{s,j}^t + e_{a,j}^t)W_K^r)^T \\
 &\quad + \frac{1}{\sqrt{d}}(p_i U_Q^r)(p_j U_K^r)^T \\
 &= \frac{1}{\sqrt{d}}h_{s,i}^{t-1}W_Q^r(W_K^r)^T(h_{a,j}^{t-1})^T + \frac{1}{\sqrt{d}}h_{s,i}^{t-1}W_Q^r(W_K^r)^T(e_{s,j}^t)^T \\
 &\quad + \frac{1}{\sqrt{d}}h_{s,i}^{t-1}W_Q^r(W_K^r)^T(e_{a,j}^t)^T + \frac{1}{\sqrt{d}}e_{s,i}^t W_Q^r(W_K^r)^T(h_{a,j}^{t-1})^T \\
 &\quad + \frac{1}{\sqrt{d}}e_{s,i}^t W_Q^r(W_K^r)^T(e_{s,j}^t)^T + \frac{1}{\sqrt{d}}e_{s,i}^t W_Q^r(W_K^r)^T(e_{a,j}^t)^T \\
 &\quad + \frac{1}{\sqrt{d}}e_{a,i}^t W_Q^r(W_K^r)^T(h_{a,j}^{t-1})^T + \frac{1}{\sqrt{d}}e_{a,i}^t W_Q^r(W_K^r)^T(e_{s,j}^t)^T \\
 &\quad + \frac{1}{\sqrt{d}}e_{a,i}^t W_Q^r(W_K^r)^T(e_{a,j}^t)^T + \frac{1}{\sqrt{d}}(p_j U_Q^r)(p_j U_K^r)^T
 \end{aligned} \tag{17}$$

In $\hat{\alpha}_{ij}^{r',t}$, r' denotes the relation of *Act to Sentiment* and t denotes the time step of recurrent dual-task reasoning. We can observe that finally, there are 10 terms in (17). The 1st term models the semantics-level interaction between node i and node j . The 2nd-9th terms model the prediction-level interactions. Specifically, the 2nd, 3rd, 4th and 7th terms model the semantics-prediction interactions; the 5th, 6th and 9th terms model the prediction-prediction interactions. The 10th term achieves relation-specific temporal modeling.

Therefore, our proposed DTR-ReTeFormer can explicitly and comprehensively model the semantics- and prediction-level interactions. And relational modeling achieves the self-task and cross-task interactions. Besides, in this process, the relation-specific temporal information is considered, facilitating dual-task reasoning.

VI. EXPERIMENTS

A. Datasets and Metrics

Datasets: We conduct experiments on two publicly available dialogue datasets: Mastodon [3] and Dailydialog [10]. The Mastodon dataset includes 269 dialogues for training and 266 dialogues for testing. And there are 3 sentiment classes and 15 act classes. Since there is no official validation set, we follow the same partition as [7]. Finally, there are 243 dialogues for training, 26 dialogues for validating, and 266 dialogues for testing. As for Dailydialog dataset, we adopt the official train/valid/test/ split from the original dataset [10]: 11,118 dialogues for training, 1,000 for validating, and 1,000 for testing. And there are 7 sentiment classes and 4 act classes.

Evaluation Metrics: Following previous works [3], [4], [7], on Dailydialog dataset, we adopt macro-average Precision (P), Recall (R), and F1 for the two tasks, while on Mastodon dataset, we ignore the neutral sentiment label in DSC task and for DAR task we adopt the average of the F1 scores weighted by the prevalence of each dialogue act.

B. Implement Details and Baselines

Both of DARER and DARER² are trained with Adam optimizer with the learning rate of $1e^{-3}$ and the batch size is 16. We exploit 300-dimensional Glove vectors for the word embeddings. And the epoch number is 100 for Mastodon and 50 for DailyDialog. Next, we introduce the different settings of other hyper-parameters for DARER and DARER².

DARER The dimension of hidden states (label embeddings) is 128 for Mastodon and 256 for DailyDialog. The step number T for recurrent dual-task reasoning is set to 3 for Mastodon and 1 for DailyDialog. The coefficient γ_s and γ_a are 3 for Mastodon and $1e^{-4}$ for DailyDialog. To alleviate overfitting, we adopt dropout, and the ratio is 0.2 for Mastodon and 0.3 for DailyDialog.

DARER² The dimension of hidden states (label embeddings) is 256 for Mastodon and 300 for DailyDialog. The step number T for recurrent dual-task reasoning is set to 5 for Mastodon and 3 for DailyDialog. For Mastodon dataset, the coefficients γ_s and γ_a are 10 and 1. For DailyDialog, the coefficients γ_s and γ_a are 0.1 and $1e^{-6}$. The dropout ratio is 0.4 for both Mastodon and DailyDialog.

For all experiments, we pick the model performing best on the validation set and then report the average results on the test set based on three runs with different random seeds. All computations are conducted on NVIDIA RTX 6000.

We compare our model with: JointDAS [3], IIIM [5], DCR-Net (Co-Attention) [4], BCDCN [6] and Co-GAT [7].

C. Main Results

1) Comparison With Baselines: Table IV lists the experiment results on the test sets of the two datasets. We can observe that:

- Our models significantly outperform all baselines, achieving new state-of-the-art (SOTA). In particular, over Co-GAT, the existing SOTA, DARER achieves an absolute improvement of 13.1% in F1 score on DSC task in

TABLE IV
EXPERIMENT RESULTS

Models	Mastodon						DailyDialog					
	DSC			DAR			DSC			DAR		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
JointDAS [3]	36.1	41.6	37.6	55.6	51.9	53.2	35.4	28.8	31.2	76.2	74.5	75.1
IIIM [5]	38.7	40.1	39.4	56.3	52.2	54.3	38.9	28.5	33.0	76.5	74.9	75.7
DCR-Net [4]	43.2	47.3	45.1	60.3	56.9	58.6	56.0	40.1	45.4	79.1	79.0	79.1
BCDCN [6]	38.2	62.0	45.9	57.3	61.7	59.4	55.2	45.7	48.6	80.0	80.6	80.3
Co-GAT [7]	44.0	53.2	48.1	60.4	60.6	60.5	65.9	45.3	51.0	81.0	78.1	79.4
Co-GAT*	45.40	48.11	46.47	62.55	58.66	60.54	58.04	44.65	48.82	79.14	79.71	79.39
DARER	56.04[†] ↑23.4%	63.33[†] ↑31.6%	59.59[†] ↑28.2%	65.08[‡] ↑4.0%	61.88[†] ↑5.5%	63.43[†] ↑4.8%	59.96[‡] ↑3.3%	49.51[†] ↑10.9%	53.42[†] ↑9.4%	81.39[†] ↑2.8%	80.80[‡] ↑1.4%	81.06[†] ↑2.1%
DARER ²	58.53[†] ↑28.9%	67.06[†] ↑39.4%	62.38[†] ↑34.2%	68.26[†] ↑9.1%	67.15[†] ↑14.5%	67.70[†] ↑11.8%	65.58[†] ↑13.0%	48.28[†] ↑8.1%	54.34[†] ↑11.3%	81.41[†] ↑2.9%	81.79[‡] ↑2.6%	81.60[†] ↑2.8%

*Denotes we reproduce the results using official code. [†] denotes that our DARER and DARER² significantly outperforms the previous best model Co-GAT with $p < 0.01$ under t-test and [‡] denotes $p < 0.05$. [†] denotes the improvement achieved by our model over Co-GAT. DARER²'s confusion matrices and F1 score on each class on Mastodon and DailyDialog test sets are illustrated in Fig. 9.

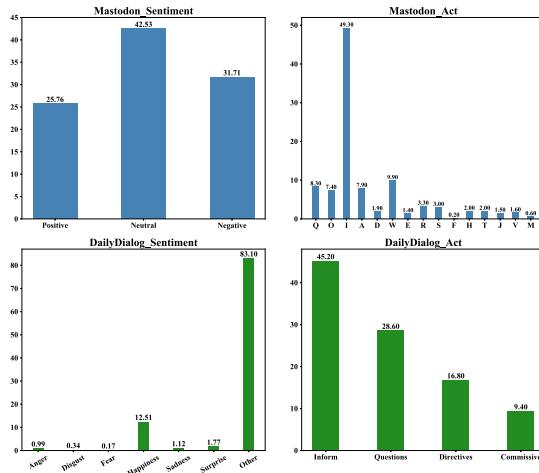


Fig. 8. Illustration of class distributions on mastodon and dailydialog datasets.

Mastodon, a relative improvement of over 28%. And DARER² achieves even larger improvements: over 34% improvement in F1 score on DSC task in Mastodon dataset. The satisfying results of DARERs come from 1) our framework integrates not only semantics-level interactions but also prediction-level interactions, thus capturing explicit dependencies other than implicit dependencies; 2) our SATG represents the speaker-aware semantic states transitions, capturing the important basic semantics benefiting both tasks; 3) our DRTG provides a rational platform on which more effective dual-task relational reasoning is conducted. 4) the advanced architecture of our DARER models allows DSC and DAR to improve each other in the recurrent dual-task reasoning process gradually.

2) DARER and DARER² show more prominent superiority on DSC task than DAR task. We surmise the probable reason is that generally, the act label is more complicated to deduce than the sentiment label in dual-task reasoning. For instance, it is easy to infer u_i 's Negative label on DSC given u_i 's Agreement label on DAR and u_{i-1} 's Negative label on DSC. Reversely, given the label information that u_i and u_{i-1} are both negative on DSC, it is hard

Ground-Truth	Mastodon Dataset (Test)	Predicted		
		Neg	Pos	Neu
Neg	271	35	66	
Pos	60	192	39	
Neu	153	104	222	
F1 score (%)	63.32	61.74	47.01	

Ground-Truth	DailyDialog Dataset (Test)	Predicted						
		No Emotion	Happiness	Surprise	Fear	Disgust	Sadness	Anger
No Emotion	5887	341	32	2	3	24	32	
Happiness	396	609	9	0	2	0	0	3
Surprise	54	9	52	0	0	0	0	1
Fear	9	0	1	7	0	0	0	0
Disgust	27	0	0	0	15	1	4	
Sadness	70	0	1	0	0	27	4	
Anger	60	5	3	0	1	1	48	
F1 score (%)	91.81	62.27	48.60	53.85	44.11	34.84	45.71	

Fig. 9. Illustration of confusion matrices and F1 score on each class on Mastodon and DailyDialog test sets. Note that on Mastodon test set, following previous works, the F1 score of the Neutral class is not counted for the final F1 score.

to infer the act label of u_i because there are several act labels possibly satisfying this case, e.g., Disagreement, Agreement, Statement.

3) Our models' improvements on DailyDialog are smaller than those on Mastodon. We speculate this is caused by the extremely unbalanced sentiment class distribution in

TABLE V
RESULTS COMPARISON BASED ON DIFFERENT PTLM ENCODERS

Models	Mastodon						
	DSC			DAR			
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	
BERT	+ Linear	61.79	61.09	60.60	70.20	67.49	68.82
	+ Co-GAT	66.03	58.13	61.56	70.66	67.62	69.08
	+ DARER	65.98	67.39	66.42	73.82	71.67	72.73
	+ DARER ²	64.47	71.10	67.61	75.34	73.04	74.17
RoBERTa	+ Linear	57.83	60.54	57.83	62.49	61.93	62.20
	+ Co-GAT	61.28	57.25	58.26	66.46	64.01	65.21
	+ DARER	61.36	67.27	63.66	70.87	68.68	69.75
	+ DARER ²	63.78	71.44	66.49	73.86	72.87	73.36
XLNet	+ Linear	61.42	67.80	63.35	67.31	63.04	65.09
	+ Co-GAT	64.01	65.30	63.71	67.19	64.09	65.60
	+ DARER	68.05	69.47	68.66	72.04	69.63	70.81
	+ DARER ²	67.20	74.24	70.42	72.45	71.47	71.96

DailyDialog. As shown in Fig. 8, in DailyDialog dataset, over 83% utterances do not express sentiment, while the act labels are rich and varied. This hinders DARER from learning valuable correlations between the two tasks.

2) *Comparison of DARER and DARER²:* From Table IV, we can find that DARER² outperforms DARER, further improving the performance. This can be attributed to the fact that the proposed SAT-ReTeFormer and DTR-ReTeFormer in DARER² can more effectively model the relational and temporal interactions than the RGCNs adopted in DARER. Especially, in DTR-ReTeFormer, since the input hidden state is superimposed with both tasks' label representation of the corresponding utterance, the relation- and structure-aware disentangled multi-head attention can explicitly and sufficiently model the relation-specific dual-task interactions, including semantics-semantics interactions, semantics-prediction interactions, and prediction-prediction interactions.

3) *Effect of Pre-Trained Language Model:* In this section, we study the effects of three PTLM encoders: BERT [43], RoBERTa [44], and XLNet [45], which replace the BiLSTM utterance encoder in the state-of-the-art model Co-GAT and our DARER models. We adopt the base versions of the PTLMs implemented in PyTorch. In our experiments, the whole models are trained by AdamW optimizer with the learning rate of $1e^{-5}$ and the batch size is 16. And the PTLMs are fine-tuned in the training process. Results are listed in Table V. We can find that since single PTLM encoders are powerful in language understanding, they obtain promising results even without any interactions between utterances or the two tasks. Nevertheless, stacking DARER on PTLM encoders further obtains around 4%-10% absolute improvements on F1. This is because our DARER models achieve relational temporal graph reasoning prediction-level interactions, which complement the high-quality semantics grasped by PTLM encoders. In contrast, Co-GAT only models the semantics-level interactions, whose advantages are diluted by the powerful PTLMs. Consequently, based on PTLM encoders, Co-GAT brings much less improvement than our DARER models.

TABLE VI
RESULTS (IN F1 SCORE) OF ABLATION EXPERIMENTS ON DARER

Variants	Mastodon		DailyDialog	
	DSC	DAR	DSC	DAR
DARER	59.59	63.43	53.42	81.06
w/o Label Embeddings	56.76	62.15	50.64	79.87
w/o $\mathcal{L}_{constraint}$	56.22	61.99	49.94	79.76
w/o SAT-RSGT	57.37	62.96	50.25	80.52
w/o DTR-RSGT	56.69	61.69	50.11	79.76
w/o TS-LSTMs	56.30	61.49	51.61	80.33
w/o Tpl Rel in SATG	58.23	62.21	50.99	80.70
w/o Tpl Rel in DRTG	57.22	62.15	50.52	80.28

D. Ablation Study

Our DARER and DARER² share the same overall model architectures, which include *label embeddings*, *constraint loss*, *SAT-RSGT*, *DTR-RSGT*, *TS-LSTMs*, *SATG* and *DRTG*. To study the effectiveness of each component, we conduct ablation experiments on DARER and Table VI lists the results.

From Table VI, we have the following observations:

- 1) Removing *label embeddings* causes prediction-level interactions not to be achieved. The sharp drops in results prove that our method of leveraging label information to achieve prediction-level interactions effectively improves dual-task reasoning via capturing explicit dependencies.
- 2) Without *constraint loss*, the two logic rules can hardly be met, so there is no constraint forcing DSC and DAR to gradually prompt each other, resulting in the dramatic decline of performances.
- 3) As the core of Dialog Understanding, *SAT-RSGT* captures speaker-aware semantic states transitions, which provides essential basic task-free knowledge for both tasks. Without it, some essential indicative semantics would be lost, then the results decrease.
- 4) The worst results of 'w/o DTR-RSGT' prove that *DTR-RSGT* is the core of DARER, and it plays a vital role in conducting dual-task relational reasoning over the semantics and label information.
- 5) The significant results decrease of 'w/o TS-LSTMs' prove that *TS-LSTMs* also plays an important role in DARER by generating task-specific hidden states for both tasks and have some capability of sequence label-aware reasoning.
- 6) Removing of the *temporal relations* (Tpl Rel)s in SATG or DRTG causes distinct results decline. This can prove the necessity and effectiveness of introducing temporal relations into dialog understanding and dual-task reasoning.

To further study the necessity of $\mathcal{L}_{constraint}$, we compare DARER² and the variant w/o $\mathcal{L}_{constraint}$ on the detailed performances of each single step, as shown in Table VII. We can observe that the $\mathcal{L}_{constraint}$ in DARER² can make the model generate better predictions at each step than w/o $\mathcal{L}_{constraint}$. Besides, thanks to the margin loss, DARER² can generate better and better predictions along the time step. However, removing $\mathcal{L}_{constraint}$ leads to fluctuations and significant drops in performance. The reason is that without the estimate loss and margin loss, the two rules cannot be integrated into the training process.

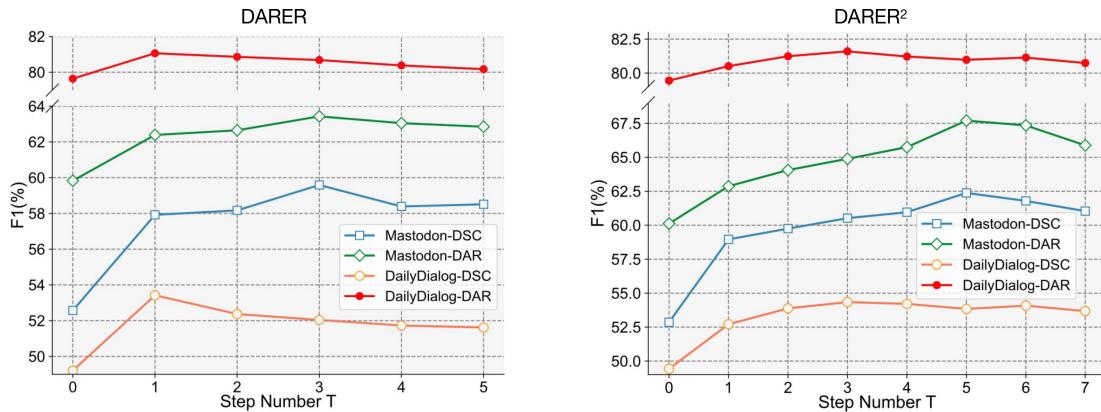
Fig. 10. Performances of DARER and DARER² over different T .

TABLE VII
THE COMPARISON OF DARER² AND $\mathcal{L}_{constraint}$ ON THE PERFORMANCES OF EACH SINGLE STEP

Model	metric	step				
		0	1	2	3	4
DARER ²	F1	57.34	59.41	61.70	62.30	62.38
	P	59.24	59.94	59.35	58.08	57.48
	R	55.65	59.29	64.96	67.19	68.36
w/o $\mathcal{L}_{constraint}$	F1	54.15	58.66	58.89	59.51	58.54
	P	58.56	62.04	55.33	57.83	55.57
	R	51.73	62.04	62.95	61.36	61.88

TABLE VIII
RESULTS (IN F1 SCORE) OF DIFFERENT SETTINGS OF RETEFORMERS

Variants	-ReTeFormer		Mastodon		DailyDialog	
	SAT	DTR	DSC	DAR	DSC	DAR
DARER ²	✓	✓	62.38	67.70	54.34	81.60
M ₁	✗	✓	61.85	66.57	54.16	81.54
M ₂	✓	✗	60.25	64.23	53.98	81.22
DARER	✗	✗	59.59	63.43	53.42	81.06

✗ denotes the corresponding ReTeFormer is replaced with the RGCN counterpart used in DARER.

Only relying on the cross-entropy loss at the final step cannot effectively improve the predictions of the previous step nor make the model generate better and better predictions along the step.

E. Superiority of ReTeFormer

In DARER² the SAT-RSGT and DTR-RSGT are achieved by our proposed SAT-ReTeFormer and DTR-ReTeFormer respectively, rather than RGCNs. To verify the superiorities of the two ReTeFormers over their RGCN counterparts in DARER, we change the setting of the two ReTeFormers and show the performances in Table VIII.

We can observe that for both SATG and DTRG, our ReTeFormer shows significant superiority over RGCN. There are two main reasons. First, our proposed ReTeFormer can conduct *fine-grained* relational temporal modeling, while RGCN can only handle the *coarse-grained* relative temporal relations. Intuitively, fine-grained relational temporal modeling can better model the latent structures of the dialog than coarse-grained one,

further benefiting dual-task reasoning. Second, the Relation-and Structure-Aware Disentangled Multi-head Attention in our proposed ReTeFormer can explicitly model the correlations between the nodes. Especially, our DTR-ReTeFormer can explicitly and comprehensively model the self-task and cross-task interactions that are of both semantics- and prediction-level.

F. Impact of Step Number T

The performances of DARER and DARER² over different T are plotted in Fig. 10. $T = 0$ denotes the output of the Initial Estimation module is regarded as final predictions. We can find that appropriately increasing T brings results improvements. Particularly, with T increasing from 0 to 1, the results increase sharply. This verifies that the Initial Estimation module can provide useful label information for dual-task reasoning. Furthermore, DARER can learn beneficial mutual knowledge from recurrent dual-task reasoning in which DSC and DAR prompt each other. Generally, when T surpasses a certain point, the performances decline slightly. The possible reason is that after the peak, more dual-task interactions cause too much deep information fusion of the two tasks, leading to the loss of some important task-specific features and overfitting.

G. Case Study

To better understand how our model works well, we compare the final predictions of Co-GAT and our DARER², as shown in Fig. 11(a). We can find that our DARER² can correctly predict all labels of both tasks, while there are some errors in Co-GAT's predictions: the act label of u_3 is incorrectly inferred as Answer, and the sentiment label of u_4 is incorrectly inferred as Negative. We suppose there are two reasons: 1) Co-GAT works on a homogeneous fully-connected dual-task graph, losing the intra- and cross-task dependencies and temporal information among the nodes; 2) Co-GAT only achieves semantics-level to implicitly models the dual-task dependencies, without incorporating prediction-level interactions.

To show how our DARER² conducts the dual-task relational temporal graph reasoning, we illustrate the dual-task reasoning process in Fig. 11(b). At step 0, the initial estimation module produces the initial label distributions. In the first step of

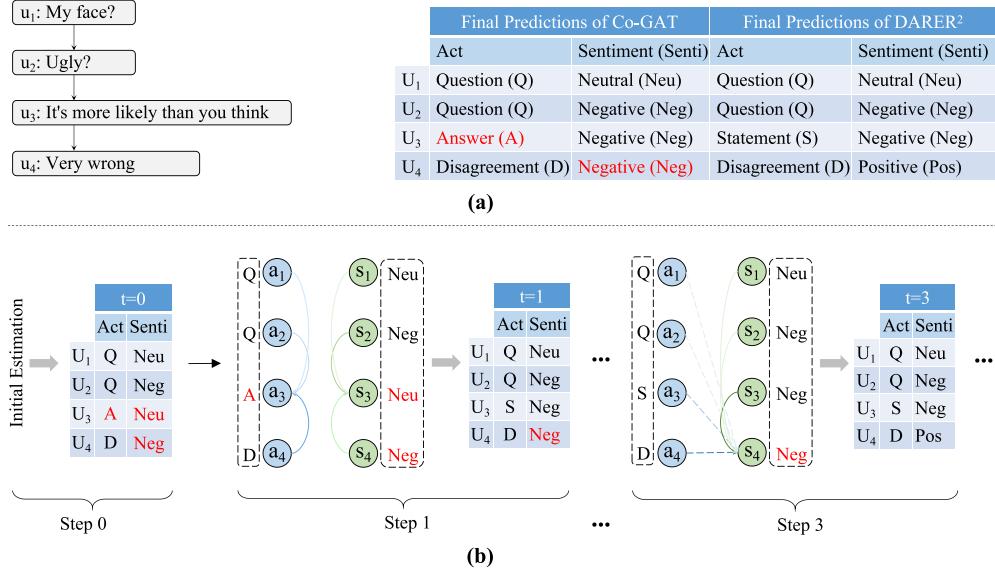


Fig. 11. Case study. (a) The example dialog and the final predictions of Co-GAT and our DARER². The red color denotes error. (b) Illustration of the estimated labels at each time step and the reasoning process. For simplification, we only list the highest probability label rather than the whole label distribution. The dashed box denotes the label estimated at the previous step. a_i and s_i denote the act node and sentiment node of u_i , respectively. The blue solid arrows denote the edges between act nodes. The green solid arrows denote the edges between sentiment nodes. The blue dashed arrows denote the edges from act nodes to sentiment nodes. Deeper color denotes a larger attention weight.

dual-task reasoning, some errors in the previously estimated labels are corrected through the intra-task interactions of act recognition task and sentiment classification task. Specifically, in act nodes, the semantics and label information of node a_4 is assigned a large weight and aggregated into node a_3 . The Disagreement label of a_4 can indicate the Statement label of a_3 . This is because in the dataset, if an utterance has a Disagreement act label, in most cases, its previous utterance has a Statement act label, which is also consistent with the real-world scenarios. In sentiment nodes, s_2 is assigned large weight and aggregated into node s_3 . Combining the semantics ‘more’ of s_3 and the Negative label of s_2 , the Negative label of s_3 can be correctly inferred. Then in the third step of dual-task reasoning, the wrong label of s_4 node is fixed. Specifically, node a_3 , a_4 and s_3 are assigned relatively large attention weights for s_4 . Regarding the labels of a_3 and a_4 , u_4 disagrees u_3 , indicating s_3 and s_4 may have opposite sentiments. And further considering the Negative label of s_4 , our model can produce the correct Positive sentiment label for u_4 . In this way, our model can gradually generate better labels through recurrent relational temporal graph reasoning.

H. Computation Efficiency

In practical application, in addition to the performance, the number of parameters, the time cost, and GPU memory required are important factors. Taking Mastodon as the testbed, we compare our DARER models with the up-to-date SOTA (Co-GAT) on these factors, and the results are shown in Table IX. Avg. F1 denotes the average of the F1 scores on the two tasks. We can find that although our DARER models surpass SOTA by a large margin, they do not significantly cost more computation

TABLE IX
COMPARISON WITH SOTA ON MODEL PARAMETERS, TRAINING TIME, GPU MEMORY REQUIRED, AND PERFORMANCE

Models	Number of Parameters ↓	Training Time per Epoch ↓	GPU ↓ Memory	Avg. F1↑
Co-GAT	3.61M	1.86s	1531MB	53.66%
DARER	2.50M	1.81s	1187MB	61.51%
Improve	-30.7%	-2.7%	-22.4%	+14.6%
DARER ²	3.83M	2.16s	1191MB	65.04%
Improve	+6.1%	+16.1%	-22.2%	+21.2%

resources. Especially, DARER is even more efficient than Co-GAT. As for DARER², although it has some more parameters and costs more training time, this is acceptable considering that it can save about 22% GPU memory and improve 21% performance. Therefore, our DARER models are relatively efficient for practical application.

I. Experiment on Joint Multiple Intent Detection and Slot Filling

To verify the generality of our method, we further conduct experiments on the task of joint multiple intent detection and slot filling.

1) *Task Definition:* The input is an utterance that can be denoted as $U = \{u_i\}_1^n$. Multiple intent detection can be formulated as a multi-label classification task that predicts multiple intents expressed in the input utterance. And slot filling is a sequence labeling task that maps each u_i into a slot label.

2) *Model Architecture:* We apply our relational temporal graph reasoning to the state-of-the-art model GL-GIN [34], forming DARER and DARER² in this joint task scenario. In

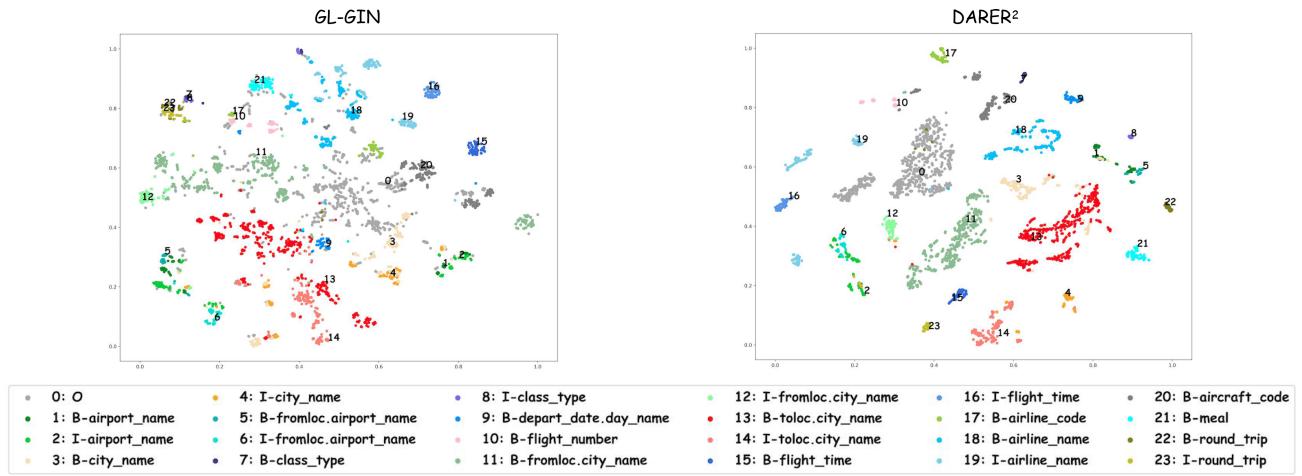


Fig. 12. Visualizations of slot hidden states generated by GL-GIN and our DARER².

GL-GIN, the dual-task graph is a semantics-label graph, which is a homogeneous graph including two groups of nodes: predicted intent label nodes and slot semantics nodes. And vanilla GAT is utilized for graph reasoning. In our DARER, the dual-task graph is a relational temporal graph, in which there are 1) intra- and cross-task relations among the two tasks’ nodes; 2) coarse-grained temporal relations among the slot semantics nodes. And RGCN is utilized for relational temporal graph reasoning. In our DARER², the dual-task graph is also a relational temporal graph, including intra- and cross-task relations. And our proposed DTR-ReTeFormer is used for relational temporal graph reasoning. Since DTR-ReTeFormer can achieve the fine-grained temporal modeling, compared with GL-GIN and DARER, DARER² can capture the dependencies between B-slot labels and their I- slot labels, and this advantage is proven in Fig. 12.

3) Datasets and Metrics. Datasets: Following previous works, the two benchmarks: MixATIS and MixSNIPS [33], [46], [47] are used as testbeds for evaluation. In MixATIS, the split of train/dev/test set is 13162/756/828 (utterances). In MixSNIPS, the split of train/dev/test set is 39776/2198/2199 (utterances).

Evaluation Metrics: Following previous works, multiple intent detection is evaluated by accuracy (Acc); slot filling is evaluated using F1 score; sentence-level semantic frame parsing is evaluated using overall Acc. Overall Acc denotes the ratio of utterances for which both intents and slots are predicted correctly.

4) Implement Details and Baselines: Following GL-GIN [34], the word and label embeddings are randomly initialized and trained with the model. The dimension of the word/label embedding is 128 on MixATIS and 256 on MixSNIPS. The dimension of the hidden state is 200. We adopt Adam [48] optimizer for model training with the default setting. For all experiments, we select the best model on the dev set and report its results on the test set.

We compare our model with Attention BiRNN [49], Slot-Gated [22], Bi-Model [50], SF-ID [24], Stack-Propagation [26], Joint Multiple ID-SF [32], AGIF [33] and GL-GIN [34].

TABLE X
RESULTS COMPARISON

Models	MixATIS			MixSNIPS		
	Overall (Acc)	Slot (F1)	Intent (Acc)	Overall (Acc)	Slot (F1)	Intent (Acc)
Attention BiRNN [49]	39.1	86.4	74.6	59.5	89.4	95.4
Slot-Gated [25]	35.5	87.7	63.9	55.4	87.9	94.6
Bi-Model [50]	34.4	83.9	70.3	63.4	90.7	95.6
SF-ID [27]	34.9	87.4	66.2	59.9	90.6	95.0
Stack-Propagation [29]	40.1	87.8	72.1	72.9	94.2	96.0
Joint Multiple ID-SF [35]	36.1	84.6	73.4	62.9	90.6	95.1
AGIF [36]	40.8	86.7	74.4	74.2	94.2	95.1
GL-GIN [37]	43.0	88.2	76.3	73.7	94.0	95.7
DARER	44.7 [†]	88.4	76.7 [†]	74.7 [†]	94.4 [†]	96.5 [†]
DARER ²	49.0 [†]	89.2 [†]	77.3 [†]	76.3 [†]	94.9 [†]	96.7 [†]

[†] Denotes our model significantly outperforms baselines with $p < 0.01$ under t-test.

5) Results and Analysis: The results comparison is shown in Table X. We can observe that our DARER models significantly outperform the state-of-the-art model GL-GIN on all datasets. In particular, DARER² significantly surpasses GL-GIN on MixATIS dataset in terms of Overall Acc, achieving 14% relative improvement. The superior performances of our DARER models verify the advantages of our proposed relation temporal graph reasoning. On one hand, relation temporal graph reasoning can effectively model the intra- and cross-task relation-specific interactions. On the other hand, it can model the temporal information among the slot semantics nodes. Especially, the DTR-ReTeFormer in DARER² can comprehensively model the fine-grained temporal information, capturing the slot dependencies. To further verify this, we visualize the slot hidden state generated by DARER² and GL-GIN, as shown in Fig. 12. We can observe that DARER²’s clusters are clearer than GL-GIN’s. Besides, the B- slot clusters of DARER² and their corresponding I- slot clusters are separated clearly. In contrast, some GL-GIN’s generated B- slot clusters and their corresponding I- slot clusters even overlap. The high quality of our DARER²’s generated hidden states can be attributed to three facts. First, GL-GIN uses the vanilla GAT for information aggregation, leading to a disadvantage: for each slot node, the different information

of the intent label nodes and other slot nodes are directly fused to it. Differently, the DTR-ReTeFormer in DARER² can achieve relation-specific information aggregation, which can better leverage the beneficial information via discriminating the contributions of the two tasks' nodes. Second, the GAT in GL-GIN cannot model the temporal information, losing the dependencies among slot nodes. Since each slot node corresponds to a word in utterance, the group of slot nodes can be regarded as a sequence, where there are temporal dependencies (e.g., I-Singer can only occur behind B-Singer). And our DARER models can achieve the relational temporal modeling, then capture the beneficial slot dependencies.

VII. CONCLUSION AND FUTURE WORK

In this paper, we present a new framework, which for the first time achieves relational temporal graph reasoning and integrates prediction-level interactions to leverage estimated label distribution as explicit and important clues other than implicit semantics. We design the SATG and DRTG to facilitate relational temporal graph reasoning of dialog understanding and dual-task reasoning. To achieve our framework, we first propose a novel model named DARER to model the relational interactions between temporal information, label information, and semantics to let two tasks gradually promote each other, which is further forced by the proposed logic-heuristic training objective. Then we propose DARER², which further enhances relational temporal graph reasoning by adopting our proposed SAT-ReTeFormer and DTR-ReTeFormer. Experimental results demonstrate the superiority of our DARER models, which surpasses previous models by a large margin in different dual-task dialog language understanding scenarios.

Our work brings two insights for dialog understanding and multi-task reasoning in dialog systems: 1) exploiting the relational temporal information of the dialog for graph reasoning; 2) leveraging estimated label distributions to capture explicit correlations between the multiple tasks. In the future, we will apply our method to other multi-task learning scenarios in dialog systems.

REFERENCES

- [1] S. Young, M. Gašić, B. Thomson, and J. D. Williams, "POMDP-based statistical spoken dialog systems: A review," *Proc. IEEE*, vol. 101, no. 5, pp. 1160–1179, May 2013.
- [2] D. Ghosal, N. Majumder, R. Mihalcea, and S. Poria, "Exploring the role of context in utterance-level emotion, act and intent classification in conversations: An empirical study," in *Proc. Findings Assoc. Comput. Linguistics: ACL-IJCNLP*, 2021, pp. 1435–1449.
- [3] C. Cerisara, S. Jafaritazehjani, A. Oluokun, and H. T. Le, "Multi-task dialog act and sentiment recognition on Mastodon," in *Proc. 29th Int. Conf. Comput. Linguistics*, 2018, pp. 745–754.
- [4] L. Qin, W. Che, Y. Li, M. Ni, and T. Liu, "DCR-net: A deep co-interactive relation network for joint dialog act recognition and sentiment classification," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 8665–8672.
- [5] M. Kim and H. Kim, "Integrated neural network model for identifying speech acts, predicates, and sentiments of dialogue utterances," *Pattern Recognit. Lett.*, vol. 101, pp. 1–5, 2018.
- [6] J. Li, H. Fei, and D. Ji, "Modeling local contexts for joint dialogue act recognition and sentiment classification with bi-channel dynamic convolutions," in *Proc. Int. Conf. Comput. Linguistics*, 2020, pp. 616–626.
- [7] L. Qin, Z. Li, W. Che, M. Ni, and T. Liu, "Co-GAT: A co-interactive graph attention network for joint dialog act recognition and sentiment classification," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 13709–13717.
- [8] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Representations*, 2018.
- [9] Y. Wang, J. Zhang, J. Ma, S. Wang, and J. Xiao, "Contextualized emotion recognition in conversation as sequence tagging," in *Proc. 23rd Annu. Meeting Special Int. Group Discourse Dialogue*, 2020, pp. 186–195.
- [10] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, and S. Niu, "DailyDialog: A manually labelled multi-turn dialogue dataset," in *Proc. Int. Joint Conf. Natural Lang. Process.*, 2017, pp. 986–995.
- [11] B. Xing and I. Tsang, "DARER: Dual-task temporal relational recurrent reasoning network for joint dialog sentiment classification and act recognition," in *Proc. Findings Assoc. Comput. Linguistics: ACL*, 2022, pp. 3611–3621.
- [12] M. S. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *Proc. Eur. Semantic Web Conf.*, 2018, pp. 593–607.
- [13] D. Hazarika, S. Poria, A. Zadeh, E. Cambria, L.-P. Morency, and R. Zimmermann, "Conversational memory network for emotion recognition in dyadic dialogue videos," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2018, pp. 2122–2132.
- [14] P. Zhong, D. Wang, and C. Miao, "Knowledge-enriched transformer for emotion detection in textual conversations," in *Proc. Conf. Empir. Methods Natural Lang. Joint Conf. Natural Lang. Process.*, 2019, pp. 165–176.
- [15] W. Jiao, M. R. Lyu, and I. King, "Real-time emotion recognition via attention gated hierarchical memory network," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 8002–8009.
- [16] L. Zhu, G. Pergola, L. Gui, D. Zhou, and Y. He, "Topic-driven and knowledge-aware transformer for dialogue emotion detection," in *Proc. Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 1571–1582.
- [17] W. Shen, S. Wu, Y. Yang, and X. Quan, "Directed acyclic graph network for conversational emotion recognition," in *Proc. Assoc. Comput. Linguistics-Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 1551–1560.
- [18] V. Raheja and J. Tetreault, "Dialogue act classification with context-aware self-attention," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2019, pp. 3727–3733.
- [19] G. Shang, A. Tixier, M. Vazirgiannis, and J.-P. Lorré, "Speaker-change aware CRF for dialogue act classification," in *Proc. Int. Conf. Comput. Linguistics*, 2020, pp. 450–464.
- [20] T. Saha, A. Patra, S. Saha, and P. Bhattacharyya, "Towards emotion-aided multi-modal dialogue act classification," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 4361–4372.
- [21] D. Hakkani-Tür et al., "Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM," in *Proc. Interspeech 2016*, 2016, pp. 715–719.
- [22] C.-W. Goo et al., "Slot-gated modeling for joint slot filling and intent prediction," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2018, pp. 753–757.
- [23] C. Li, L. Li, and J. Qi, "A self-attentive model with gate mechanism for spoken language understanding," in *Proc. Conf. Empir. Methods Natural Lang. Joint Conf. Natural Lang. Process.*, 2018, pp. 3824–3833.
- [24] H. E. P. Niu, Z. Chen, and M. Song, "A novel bi-directional interrelated model for joint intent detection and slot filling," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 5467–5471.
- [25] Y. Liu, F. Meng, J. Zhang, J. Zhou, Y. Chen, and J. Xu, "CM-Net: A novel collaborative memory network for spoken language understanding," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, Hong Kong, China, 2019, pp. 1051–1060.
- [26] L. Qin, W. Che, Y. Li, H. Wen, and T. Liu, "A stack-propagation framework with token-level intent detection for spoken language understanding," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 2078–2087.
- [27] C. Zhang, Y. Li, N. Du, W. Fan, and P. Yu, "Joint slot filling and intent detection via capsule neural networks," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 5259–5267.
- [28] D. Wu, L. Ding, F. Lu, and J. Xie, "SlotRefine: A fast non-autoregressive model for joint intent detection and slot filling," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 1932–1937.
- [29] B. Xing and I. Tsang, "Group is better than individual: Exploiting label topologies and label relations for joint multiple intent detection and slot filling," in *Proc. Conf. on Empirical Methods Natural Lang. Process.*, Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022, pp. 3964–3975.

- [30] B. Xing and I. Tsang, "Co-guiding net: Achieving mutual guidances between multiple intent detection and slot filling via heterogeneous semantics-label graphs," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022, pp. 159–169.
- [31] B. Kim, S. Ryu, and G. G. Lee, "Two-stage multi-intent detection for spoken language understanding," *Multimedia Tools Appl.*, vol. 76, no. 9, pp. 11 377–11 390, 2017.
- [32] R. Gangadharaiyah and B. Narayanaswamy, "Joint multiple intent detection and slot labeling for goal-oriented dialog," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2019, pp. 564–569.
- [33] L. Qin, X. Xu, W. Che, and T. Liu, "AGIF: An adaptive graph-interactive framework for joint multiple intent detection and slot filling," in *Proc. Findings Assoc. Comput. Linguistics: Empirical Methods Natural Lang. Process.*, 2020, pp. 1807–1816.
- [34] L. Qin, F. Wei, T. Xie, X. Xu, W. Che, and T. Liu, "GL-GIN: Fast and accurate non-autoregressive model for joint multiple intent detection and slot filling," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 178–188.
- [35] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [36] S. Zheng, F. Wang, H. Bao, Y. Hao, P. Zhou, and B. Xu, "Joint extraction of entities and relations based on a novel tagging scheme," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 1227–1236.
- [37] B. Xing and I. Tsang, "DigNet: Digging clues from local-global interactive graph for aspect-level sentiment classification," 2022, *arXiv:2201.00989*.
- [38] B. Xing and I. W. Tsang, "Understand me, if you refer to aspect knowledge: Knowledge-aware gated recurrent memory network," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, no. 5, pp. 1092–1102, Oct. 2022.
- [39] B. Xing and I. W. Tsang, "Out of context: A new clue for context modeling of aspect-based sentiment analysis," *J. Artif. Intell. Res.*, vol. 74, pp. 627–659, 2022.
- [40] B. Xing and I. W. Tsang, "Co-evolving graph reasoning network for emotion-cause pair extraction," 2023, *arXiv:2306.04340*.
- [41] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
- [42] G. Ke, D. He, and T.-Y. Liu, "Rethinking positional encoding in language pre-training," in *Proc. Int. Conf. Learn. Representations*, 2021.
- [43] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2019, pp. 4171–4186.
- [44] Y. Liu et al., "RoBERTa: A robustly optimized bert pretraining approach," 2019, *arXiv: 1907.11692*.
- [45] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 5753–5763.
- [46] C. T. Hemphill, J. J. Godfrey, and G. R. Doddington, "The ATIS spoken language systems pilot corpus," in *Proc. Speech Natural Lang.: Proc. Workshop Held*, 1990, pp. 96–101.
- [47] A. Coucke et al., "Snips voice platform: An embedded spoken language understanding system for private-by-design voice interfaces," 2018, *arXiv: 1805.10190*.
- [48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.
- [49] B. Liu and I. Lane, "Attention-based recurrent neural network models for joint intent detection and slot filling," in *Proc. Interspeech*, 2016, pp. 685–689.
- [50] Y. Wang, Y. Shen, and H. Jin, "A Bi-model based RNN semantic frame parsing model for intent detection and slot filling," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2018, pp. 309–314.



Bowen Xing received the BE degree and master degree from the Beijing Institute of Technology, Beijing, China, in 2017 and 2020, respectively. He is currently working toward the third-year PhD degree with the Australian Artificial Intelligence Institute (AAII), University of Technology Sydney (UTS). His research focuses on graph neural networks, multi-task learning, sentiment analysis, and dialog system.



Ivor W. Tsang (Fellow, IEEE) is the director of A*STAR Centre for Frontier AI Research (CFAR). Previously, he was a professor of artificial intelligence, with University of Technology Sydney (UTS), and research director of the Australian Artificial Intelligence Institute (AAII). His research focuses on transfer learning, deep generative models, learning with weakly supervision, Big Data analytics for data with extremely high dimensions in features, samples and labels. His work is recognised internationally for its outstanding contributions to those fields. In 2013,

Prof Tsang received his ARC Future Fellowship for his outstanding research on Big Data analytics and large-scale machine learning. In 2019, his JMLR paper "Towards ultrahigh dimensional feature selection for Big Data" received the International Consortium of Chinese Mathematicians Best Paper Award. In 2020, he was recognized as the AI 2000 AAAI/JCIAI Most Influential Scholar in Australia for his outstanding contributions to the field, between 2009 and 2019. His research on transfer learning was awarded the Best Student Paper Award with CVPR 2010 and the 2014 IEEE TMM Prize Paper Award. In addition, he received the IEEE TNN Outstanding 2004 Paper Award, in 2007 for his innovative work on solving the inverse problem of non-linear representations. Recently, Prof Tsang was conferred the IEEE Fellow for his outstanding contributions to large-scale machine learning and transfer learning. Prof Tsang serves as the Editorial Board for the *Journal of Machine Learning Research*, *The Madras Law Journal*, *Journal of Artificial Intelligence Research*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Artificial Intelligence*, *IEEE Transactions on Big Data*, and *IEEE Transactions on Emerging Topics in Computational Intelligence*. He serves as a senior area chair/area chair for NeurIPS, ICML, AAAI and IJCAI, and the steering committee of ACM.