Dynamic Schema Graph Fusion Network for Multi-Domain Dialogue State Tracking

Anonymous ACL submission

Abstract

Dialogue State Tracking (DST) aims to keep track of users’ intentions during the course of a conversation. In DST, modelling the relations among domains and slots is still an under-studied problem. Existing approaches that have considered such relations generally fall short in: (1) fusing prior slot-domain membership relations and dialogue-aware dynamic slot relations explicitly, and (2) generalizing to unseen domains. To address these issues, we propose a novel Dynamic Schema Graph Fusion Network (DSGFNet), which generates a dynamic schema graph to explicitly fuse the prior slot-domain membership relations and dialogue-aware dynamic slot relations. It also uses the schemata to facilitate knowledge transfer to new domains. DSGFNet consists of a dialogue utterance encoder, a schema graph encoder, a dialogue-aware schema graph evolving network, and a schema graph enhanced dialogue state decoder. Empirical results on benchmark datasets, including SGD, MultiWOZ2.1, and MultiWOZ2.2, show that DSGFNet outperforms the existing methods.

1 Introduction

Task-oriented dialogue systems can help users accomplish different tasks (Huang et al., 2020), such as flight reservation, food ordering, and appointment scheduling. Conventionally, task-oriented dialogue systems consist of four modules (Zhang et al., 2020c): natural language understanding (NLU), dialogue state tracking (DST), dialogue manager (DM), and natural language generation (NLG). In this paper, we will focus on the DST module. The goal of DST is to extract users’ goals or intentions as dialogue states and keep these states updated over the whole dialogue. In order to track users’ goals, we need to have a predefined domain knowledge referred to as a schema, which consists of slot names and their descriptions. Figure 1 gives an example of DST in a sample dialogue.

Many models have been developed for DST due to its importance in task-oriented dialogue systems. Traditional approaches use deep neural networks or pre-trained language models to encode the dialogue context and infer slot values from it (Zhong et al., 2018; Ramadan et al., 2018; Wu et al., 2019; Ren et al., 2019; Zhang et al., 2020a; Hu et al., 2020; Gao et al., 2020; Zhang et al., 2020a,b). These models predict slot values without considering the relations among domains and slots. However, domains and slots in a dialogue are unlikely to be entirely independent, and ignoring the relations among domains and slots may lead to sub-optimal performance. To address this issue, several recent works have been proposed to model the relations among domains and slots in DST. Some of them introduce...
predefined schema graphs to incorporate prior slot-domain membership relations, which are defined based on human experience in advance (Chen et al., 2020; Zhu et al., 2020). The others use an attention mechanism to capture dialogue-aware dynamic slot relations (Feng et al., 2021; Heck et al., 2020). The dialogue-aware dynamic relations are the logical relations of slots across domains, which are highly related to specific dialogue contexts.

However, existing DST models that involve the relations among domains and slots suffer from two major issues: (1) They fail to fuse the prior slot-domain membership relations and dialogue-aware dynamic slot relations explicitly; and (2) They fail to consider their generalizability to new domains.

In practical scenarios, task-oriented dialogue systems need to support a large and constantly increasing number of new domains.

To tackle these issues, we propose a novel approach named DSGFNet (Dynamic Schema Graph Fusion Network). For the first issue, DSGFNet dynamically updates the schema graph consisting of the predefined slot-domain membership relations with the dialogue-aware dynamic slot relations. To incorporate the dialogue-aware dynamic slot relations explicitly, DSGFNet adds three new edge types to the schema graph: co-reference relations, co-update relations, and co-occurrence relations. For the second issue, to improve its generalizability, DSGFNet employs a unified model containing schema-agnostic parameters to make predictions.

Specifically, our proposed DSGFNet comprises of four components: a BERT-based dialogue utterance encoder to contextualize the current turn dialogue context and history, a BERT-based schema graph encoder to generalize to unseen domains and model the prior slot-domain membership relations on the schema graph, a dialogue-aware schema graph evolving network to augment the dialogue-aware dynamic slot relations on the schema graph, and a schema graph enhanced dialogue state decoder to extract value spans from the candidate elements considering the evolved schema graph.

The contributions of this paper can be summarized as follows:

- We improve DST by proposing a dynamic, explainable, and general schema graph which explicitly models the relations among domains and slots based on both prior knowledge and the dialogue context, no matter whether the domains and slots are seen or not.

- We develop a fusion network, DSGFNet, which effectively enhances DST generating a schema graph out of the combination of prior slot-domain membership relations and dialogue-aware dynamic slot relations.

- We conduct extensive experiments on three benchmark datasets (i.e., SGD, MultiWOZ2.1, and MultiWOZ2.2) to demonstrate the superiority of DSGFNet and the importance of the relations among domains and slots in DST.

2 Related Work

Recent DST approaches mainly focus on encoding the dialogue contexts with deep neural networks (e.g., convolutional and recurrent networks) and inferring the values of slots independently (Zhong et al., 2018; Ramadan et al., 2018; Wu et al., 2019; Ren et al., 2019; Zhang et al., 2020a; Hu et al., 2020; Gao et al., 2020). With the prevalence of pre-trained language models, such as BERT (Devlin et al., 2019) and GPT-2 (Radford et al., 2019), a great variety of DST approaches have been developed on top of these pre-trained models (Zhang et al., 2020a,b; Lin et al., 2020). The relations among domains and slots are not considered in the above approaches. However, the prior slot-domain membership relations can facilitate the sharing of domain knowledge and the dialogue-aware dynamic slot relations can conduct dialogue history understanding. Ignoring these relations may lead to sub-optimal performance.

To fill in this gap, several new DST approaches, which involve the relations among domains and slots, have been proposed. Some of them leverage a graph structure to capture the slot-domain membership relations (Lin et al., 2021; Chen et al., 2020; Zhu et al., 2020; Zeng and Nie, 2020; Ouyang et al., 2020). Specifically, a predefined schema graph is employed to represent the slot-domain membership relations. However, they fail to incorporate the dialogue-aware dynamic slot relations into the schema graph. The other approaches utilize the attention mechanism to learn dialogue-aware dynamic slot relation features in order to facilitate information flow among slots (Zhou and Small, 2019; Feng et al., 2021; Heck et al., 2020; Hu et al., 2020; Ye et al., 2021). However, these approaches ignore the slot-domain membership relations defined by prior knowledge. Since both the prior slot-domain membership relations and dialogue-aware dynamic
slot relations can enhance DST performance, our approach is developed to combine them in an effective way.

Given that a deployed dialogue system may encounter an ever-increasing number of new domains that have limited training data available, the DST module should be capable of generalizing to unseen domains. Recent DST approaches have focused on using zero-shot learning to achieve this goal (Rastogi et al., 2020; Noroozi et al., 2020). These approaches exploit the natural language descriptions of schemata to transfer knowledge across domains. However, they ignore the relations among domains and slots. In this work, we propose a unified framework to fuse the prior slot-domain membership relations and dialogue-aware dynamic slot relations, no matter whether the domains are seen or not.

3 Dynamic Schema Graph Fusion Network

The proposed DSGFNet consists of four components: (1) a BERT-based dialogue utterance encoder that aims to contextualize the tokens of the current turn and the dialogue history; (2) a schema graph encoder that is able to generalize to unseen domains and shares information among predefined slot-domain membership relations; (3) a dialogue-aware schema graph evolving network that adds the dialogue-aware dynamic slot relations into the schema graph; and (4) a schema graph enhanced dialogue state decoder that extracts the value span from the candidate elements based on the evolved schema graph. Figure 2 illustrates the architecture of DSGFNet.

3.1 Dialogue Utterance Encoder

This encoder takes as input the current and previous dialogue utterances. Specifically, the input is a sequence of tokens with length $K$, i.e., $[t_1, ..., t_K]$. Here, we set the first token $t_1$ to \[CLS\]; subsequent are the tokens in the current dialogue utterance and the ones in the previous dialogue utterances, which are separated by \[SEP\]. We employ BERT (Devlin et al., 2019) to obtain contextual token embeddings. The output is a tensor of all the token embeddings $B = [b_1, ..., b_K]$, with one embedding for each token.

3.2 Schema Graph Encoder

To make use of the slot-domain membership relations defined by prior domain knowledge, we construct a schema graph based on the predefined ontology. An example is shown in Figure 2. In this schema graph, each node represents either a domain or a slot, and all the slot nodes are connected to their corresponding domain nodes. In order to allow information propagation across domains, all the domain nodes are connected with each other.

Schema-Agnostic Embedding Initializer. To generalize to unseen domains, DSGFNet initializes the schema graph node embeddings via a schema-agnostic projection. Inspired by zero-shot learning (Romera-Paredes and Torr, 2015), we propose a schema-agnostic embedding initializer to project schemata across domains into a unified semantic distribution. Specifically, we feed the natural language descriptions of slots and domains into BERT...
to obtain the semantic embeddings for all slots and domains $I = \{i_1, \ldots, i_{N+M}\}$, where $N$ and $M$ are the number of slots and domains, respectively. We constrain the schema embedding initializer not to have any domain-specific parameters so that it can generalize to unseen domains.

**Slot-Domain Membership Reasoning Network.** To involve the prior slot-domain membership relations into the schema graph node embeddings, DSGFNet propagates information among slots and domains over the schema graph. We add a self-loop to each node because the nodes need to propagate information to themselves. Inspired by the GAT model (Veličković et al., 2018), we propose a slot-domain membership relation reasoning network to propagate information over the schema graph. For each node, we first compute attention scores $\alpha$ for its neighbours. These attention scores are used to weigh the importance of each neighboring node. Formally, the attention scores are calculated as follows:

$$h_{i,j} = \text{ReLU}(W^T \cdot [i_i, i_j]),$$  
$$\alpha_{i,j} = \frac{\exp(h_{i,j})}{\sum_{k \in \mathcal{N}_i} \exp(h_{i,k})},$$  

where $W$ is a matrix of parameters and $\mathcal{N}_i$ is the neighborhood of the $i$-th node. The normalized attention coefficients and the activation function are used to compute a non-linear weighted combination of the neighbours. This is used to compute the tensor of the schema graph node embeddings $G = (g_1, \ldots, g_{N+M})$:

$$g_i = \text{ReLU} \left( \sum_{j \in \mathcal{N}_i} \alpha_{i,j} \cdot i_j \right),$$

where $i \in \{1, \ldots, N+M\}$. To explore the higher-order connectivity information of slots across domains, we stack $l$ layers of the reasoning network. Each layer takes the node embeddings from the previous layer as input, and outputs the updated node embeddings to the next layer.

**3.3 Schema Graph Evolving Network**

We propose a schema graph evolving network to incorporate the dialogue-aware dynamic slot relations into the schema graph, which is composed of two layers, a schema-dialogue fusion layer and a dynamic slot relation completion layer.

**Schema-Dialogue Fusion Layer.** Since the dynamic slot relations are related to the dialogue context, we need to fuse the dialogue context information into the schema graph. We adopt the multi-head attention (Vaswani et al., 2017) to achieve this goal. The mathematical formulation is:

$$H = \text{MultiHead}(Q = g_i, K = B, V = B),$$

$$\tilde{g}_i = H \cdot W_a,$$

where $W_a$ is learnable parameters of a linear projection after the multi-head attention, and $\tilde{g}_i$ is the dialogue-aware schema graph node embeddings.

**Dynamic Slot Relation Completion Layer.** This layer aims to augment the dynamic slot relations on the schema graph based on the dialogue-aware node embeddings. To involve the dialogue-aware dynamic slot relations into DST explicitly, DSGFNet defines three types of dynamic slot relations: (1) Co-reference relations occur when a slot value has been mentioned earlier in the dialogue and has been assigned to another slot; (2) Co-update relations occur when slot values are updated together at the same dialogue turn, and; (3) Co-occurrence relations occur when slots with a high co-occurrence probability in a large dialogue corpus appear together in the current dialogue. Specifically, we feed the dialogue-aware slot node representations into a multi-layer perceptron followed by a 4-way softmax function to identify the relations between slot pairs, which include the none relation and the three dynamic relations mentioned above. Formally, given the $i$-th and $j$-th dialogue-aware slot node embeddings $\tilde{g}_i$ and $\tilde{g}_j$, we obtain an adjacent matrix of the dynamic slot relations for all slot pairs as follows:

$$A(i, j) = \arg \max \left( \text{softmax}(\text{MLP}(\tilde{g}_i \oplus \tilde{g}_j)) \right).$$

With $A$, we add dynamic slot relation edges to the schema graph.

**3.4 Dialogue State Decoder**

To decode the slot values by means of incorporating the slot-domain membership relations and dialogue-aware dynamic slot relations which are captured by the evolved schema graph, we propose a schema graph enhanced dialogue state decoder.

To learn a more comprehensive slot node embedding, we need to fuse multiple relations on the evolved schema graph. DSGFNet divides different relations on the schema graph into sub-graphs $R_s, R_r, R_u, R_o$, which represent slot-domain membership relation, co-reference relation, co-update.
We conduct experiments on three task-oriented dialogue benchmark datasets: SGD (Rastogi et al., 2020), MultiWOZ2.2 (Zang et al., 2020), and MultiWOZ2.1 (Eric et al., 2020). Among them, SGD is by far the most challenging dataset which contains over 16,000 conversations between a human user and a virtual assistant across 16 domains. In particular, it also includes unseen domains in the test set. MultiWOZ2.2 and MultiWOZ2.1 are smaller human-human conversations benchmark datasets, which contain over 8,000 multi-turn dialogues across 8 and 7 domains, respectively. MultiWOZ2.2 is a revised version of MultiWOZ2.1, which is re-annotated with a different set of inter-annotators and also canonicalized entity names. Statistics about the datasets are provided in Table 1.

### 4.2 Baselines

We make a comparison with the following existing models, which are divided into two categories, predicting the dialogue state independent of the relations among domains and slots, or based on such relations. The methods ignoring the relations among domains and slots are: TRADE (Wu et al., 2019), a generation model which generates dialogue states from utterances using a copy mechanism; DS-DST (Zhang et al., 2020a), a dual strategy that classifies over a picklist or finding values from a slot span; SOM-DST (Kim et al., 2020), a selectively overwriting mechanism which first predicts state operation on each of the slots and then overwrites with new values; MinTL-BART (Lin et al., 2020), a plug-and-play pre-trained model which jointly learns dialogue state tracking and dialogue response generation; SGD-baseline (Rastogi et al., 2020), a schema-guided paradigm that predicts states for unseen domains, and; FastSGT (Noroozi et al., 2020), a BERT-based model that uses multi-head attention projections to analyze dialogue history; PPTOD (Su et al., 2021), a multi-task pre-training strategy that allows the model to learn the primary TOD task completion skills from heterogeneous dialog corpora. The methods incorporating the relations among domains and slots include: SST (Chen et al., 2020), a graph model.
which fuses information from utterances and static schema graph; TripPy (Heck et al., 2020), an open-vocabulary model which copies values from dialogue context, or slot values in previous dialogue state, and; Seq2Seq-DU (Feng et al., 2021), a sequence-to-sequence framework which decodes dialogue states in a flatten format.

4.3 Evaluation Measures

Our evaluation metrics are consistent with prior works on these datasets. We compute the Joint Goal Accuracy (Joint GA) on all test sets for straightforward comparison with the state-of-the-art methods. Joint GA is defined as the ratio of dialogue turns for which all slots have been filled with the correct values according to the ground truth.

4.4 Experimental Settings

We use the pre-trained BERT model ([BERT-Base, Uncased]) to encode utterances and schema descriptions. The BERT models are fine-tuned in the training process. The maximum length of an input sequence is set to 512. The hidden size of the schema graph encoder and the schema graph evolving network is set to 256. The dropout probability is 0.3. The balance coefficient λ is 0.5. Adam (Kingma and Ba, 2014) is used for optimization with an initial learning rate (LR) of 2e-5.

We conduct training with a warm-up proportion of 10% and let the LR decay linearly after the warm-up phase. The effects of some crucial parameters are shown in Appendix A.

5 Results and Discussion

Tables 2, 3, 4 show the performance of DSGFNet as well as the baselines on three datasets respectively. It is shown that DSGFNet achieves state-of-the-art performance on SGD, MultiWOZ2.2. And the performance on MultiWOZ2.1 are comparable with the state-of-the-art. Most notably, DSGFNet improves the performance on SGD most significantly, which has the most complex schemata, compared to the runner-up. This demonstrates the success of the dynamic schema graph in DSGFNet. The more plentiful the relations among domains and slots are, the better performance DSGFNet can achieve. The following analysis provides a better understanding of our model’s strengths.

5.1 Ablation Study

We conduct an ablation study on DSGFNet to quantify the contributions of various factors: the usage of slot-domain membership relations, dynamic slot relations, and multiple relation aggregation. The results indicate that the dynamic schema graph of DSGFNet is indispensable for DST.

### Effect of Slot-Domain Membership Relations

To check the effectiveness of the slot-domain membership relations, we remove the schema graph by replacing the prior slot-domain relation adjacency matrix with an identity matrix I. Results in Table 5 show that the joint goal accuracy of DSGFNet without the slot-domain membership relations decreases markedly on SGD, MultiWOZ2.2,
and MultiWOZ2.1. It indicates that the schema graph, which contains the slot-domain membership relations, can facilitate knowledge sharing among domains and slots to enhance DST.

**Effect of Dynamic Slot Relations**

To investigate the effectiveness of the dialogue-aware dynamic slot relations in the schema graph, we eliminate the evolving network of DSGFNet. Table 5 shows the results on SGD, MultiWOZ2.2, and MultiWOZ2.1 in terms of joint goal accuracy. One can observe that without the dynamic slot relations the performance deteriorates considerably. In addition, there is a more markedly performance degradation compared with the results of the slot-domain membership relations. It indicates that the dynamic slot relations are more essential for DST, which can facilitate the understanding of the dialogue context.

**Effect of Multiple Relation Aggregation**

To validate the effectiveness of the schema graph relation aggregation mechanism in the dialogue state decoder, we directly concatenate all sub-graph representations instead of calculating a weighted sum via the sub-graph attention. As shown in Table 5, the performance of the models without the relation aggregation layer in terms of joint goal accuracy decreases markedly compared to DSGFNet. It indicates that the attentions to different types of relations affect the dialogue understanding ability.

### 5.2 Further Analysis

**Prediction of Dynamic Slot Relations**

In order to test the discriminative capability of DSGFNet for dynamic slot relations, we evaluate the performance of the schema graph evolving network. Since baselines cannot predict the dynamic slot relations explicitly, we compare DSGFNet with the BERT-based classification approach. Following the classification task in BERT, the input sequence starts with [CLS], followed by the tokens of the dialogue context and slot pairs, separated by [SEP], and the [CLS] representation is fed into an output layer for classification. Figure 3 shows the results on SGD, MultiWOZ2.2, and MultiWOZ2.1 in terms of F1 and Accuracy. From the results, we observe that DSGFNet outperforms BERT significantly. We conjecture that it is due to the exploitation of schema graph with slot-domain membership relations in DSGFNet. In addition, since BERT without schema encoder cannot solve unseen domains, there is a significant performance degradation on SGD which contains a large number of unseen domains in the test set.

### Table 6: Performance comparison of DSGFNet with different dynamic slot relations on SGD, MultiWOZ2.2 and MultiWOZ2.1 datasets.

<table>
<thead>
<tr>
<th>Model</th>
<th>Joint GA SGD</th>
<th>Joint GA MultiWOZ2.2</th>
<th>Joint GA MultiWOZ2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>w All Dynamic Relations</td>
<td>32.1%</td>
<td>55.3%</td>
<td>56.7%</td>
</tr>
<tr>
<td>w Co-reference Relation</td>
<td>29.8%</td>
<td>53.9%</td>
<td>54.7%</td>
</tr>
<tr>
<td>w Co-occurrence Relation</td>
<td>31.7%</td>
<td>53.3%</td>
<td>55.9%</td>
</tr>
<tr>
<td>w Co-update Relation</td>
<td>30.1%</td>
<td>53.5%</td>
<td>54.5%</td>
</tr>
<tr>
<td>w/o Dynamic Relations</td>
<td>28.6%</td>
<td>52.2%</td>
<td>53.2%</td>
</tr>
</tbody>
</table>

**Effects of Each Type of Dynamic Slot Relation**

To better illustrate the effectiveness of augmenting slot relations on the schema graph, we study how different dynamic slot relations affect the DST performance. Table 6 presents the joint goal accuracy of DSGFNet with different dynamic relations on SGD, MultiWOZ2.2, and MultiWOZ2.1. One can see that the performance of DSGFNet with each type of dynamic slot relation surpasses that without any dynamic slot relations considerably. Thus, all types of dynamic slot relations in the schema graph are helpful for dialogue understanding. Furthermore, the performance of DSGFNet with co-occurrence relation is superior to the performance with the other two dynamic slot relations. We conjecture that it is due to the fact that a large percentage of dynamic relations is the co-occurrence relation, which has an incredible effect on DST.

**Dealing with Unseen Domains**

We analyze the generalization ability of DSGFNet. Table 8 shows the joint goal accuracy of DSGFNet in different domains on SGD. (Note that only SGD
We make qualitative analysis on the results of DSGFNet and Seq2Seq-DU on SGD. We find that DSGFNet can make a more accurate inference of dialogue states by using the dynamic schema graph. For example, as shown in Table 7, “city”-“location” is predicted as co-reference relation, “city”-“date” and “number of seats”-“ride type” are predicted as co-update relation, “city”-“date” is predicted as co-occurrence relation. Based on the dynamic schema graph, DSGFNet propagates information involving slot-domain membership relations and dynamic slot relations. Thus, it infers slot values more correctly. In contrast, since Seq2seq-DU ignores the dynamic slot relations, it cannot properly infer the values of “location” and “ride type”, which have dynamic slot relations with other slots.

### 6 Conclusion

We have proposed a new approach to DST, referred to as DSGFNet, which effectively fuses prior slot-domain membership relations and dialogue-aware dynamic slot relations on the schema graph. To incorporate the dialogue-aware dynamic slot relations into DST explicitly, DSGFNet identifies co-reference, co-update, and co-occurrence relations. To improve the generalization ability, DSGFNet employs a schema-agnostic graph attention network to share information. Experimental results show that DSGFNet outperforms the existing methods in DST on three benchmark datasets, including SGD, MultiWOZ2.1, and MultiWOZ2.2. For future work, we intend to further enhance our approach by utilizing more complex schemata and data augmentation techniques.
References


Yue Feng, Yang Wang, and Hang Li. 2021. A sequence-to-sequence approach to dialogue state tracking. In ACL.

Shuyang Gao, Sanchit Agarwal, Tagyoung Chung, Di Jin, and Dilek Hakkani-Tür. 2020. From machine reading comprehension to dialogue state tracking: Bridging the gap. In ACL.


Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020. Challenges in building intelligent open-domain dialog systems. In TOIS.


Osman Ramadan, Pawel Budzianowski, and Milica Gašić. 2018. Large-scale multi-domain belief tracking with knowledge sharing. In ACL.


Liliang Ren, Jianmo Ni, and Julian McAuley. 2019. Scalable and accurate dialogue state tracking via hierarchical sequence generation. In EMNLP.


Fanghua Ye, Jarana Manotumruksa, Qiang Zhang, Shenghui Li, and Emine Yilmaz. 2021. Slot self-attentive dialogue state tracking. In WWW.


Yichi Zhang, Zhijian Ou, Huixin Wang, and Junlan Feng. 2020b. A probabilistic end-to-end task-oriented dialog model with latent belief states towards semi-supervised learning. In EMNLP.

Zheng Zhang, Ryuichi Takanobu, Qi Zhu, Minlie Huang, and Xiaoyan Zhu. 2020c. Recent advances and challenges in task-oriented dialog systems. In Science China Technological Sciences.


Li Zhou and Kevin Small. 2019. Multi-domain dialogue state tracking as dynamic knowledge graph enhanced question answering. In NIPS.


### A Analysis of Parameters in DSGFNet

We further investigate the impacts of parameter settings on the performance of DSGFNet on SGD, MultiWOZ2.2, and MultiWOZ2.1. We validate the effects of four factors: the layer of propagation on the schema graph, the number of selected dialogue turns used in the schema-dialogue fusion layer, the layer of MLP in the dynamic slot relation completion layer, and the balance coefficient $\lambda$ in the loss function. Figures 4, 5, 6, 7 show the results of DSGFNet with varying parameters on SGD, MultiWOZ2.2, and MultiWOZ2.1 in terms of joint goal accuracy. We observe that the optimal layer of propagation is not consistent across datasets. It seems that 3 is desired in more datasets. In addition, DSGFNet demonstrates the best performance when leveraging full dialogue history. We conjecture that it is due to that the incomplete dialogue history leads to confusing information. Moreover, 8 layers MLP for relation completion obtains the optimal performance over three datasets. Furthermore, the optimal performance is consistently achieved when the balance coefficient $\lambda$ is around 0.5.

![Figure 4: Performance comparison w.r.t. the layer of propagation on the schema graph.](image)

![Figure 5: Performance comparison w.r.t. the number of dialogue turns used in the schema-dialogue fusion layer.](image)

![Figure 6: Performance comparison w.r.t. the layer of MLP in the dynamic slot relation completion layer.](image)
Figure 7: Performance comparison \textit{w.r.t.} the balance coefficient in the loss function.