Exploiting domain-slot related keywords description for Few-Shot Cross-Domain Dialogue State Tracking

Qixiang Gao1∗, Guanting Dong1∗, Yutao Mou1∗, Liwen Wang1
Chen Zeng1, Daichi Guo1, Mingyang Sun1, Weiran Xu1∗
1Beijing University of Posts and Telecommunications, Beijing, China
{gqx,dongguanting,myt,w_liwen}@bupt.edu.cn
{chenzeng,guodaichi,mysun,xuweiran}@bupt.edu.cn

Abstract

Collecting dialogue data with domain-slot-value labels for dialogue state tracking (DST) could be a costly process. In this paper, we propose a novel framework based on domain-slot related description to tackle the challenge of few-shot cross-domain DST. Specifically, we design an extraction module to extract domain-slot related verbs and nouns in the dialogue. Then, we integrates them into the description, which aims to prompt the model to identify the slot information. Furthermore, we introduce a random sampling strategy to improve the domain generalization ability of the model. We utilize a pre-trained model to encode contexts and description and generates answers with an auto-regressive manner. Experimental results show that our approaches substantially outperform the existing few-shot DST methods on MultiWOZ and gain strong improvements on the slot accuracy comparing to existing slot description methods.

1 Introduction

Dialogue state tracking (DST) is an essential component in a task-oriented dialogue system. It aims to keep track of users’ domains, intents and slots information at each turn of the conversation, which helps to provide sufficient information for selecting the next system operation (Balaraman et al., 2021). Recent neural-based DST models (Wu et al., 2019; Heck et al., 2020; Feng et al., 2020; Rastogi et al., 2020; Ye et al., 2021) have made significant progress under the availability of large-scale labeled data. However, due to the high cost of data annotation and the lack of sufficient data in some specific domains, the performance of the general model will drop significantly. Therefore, lack of generalization to new domains hinder the further application of task-oriented dialogue systems in practical industrial scenarios.

∗The first three authors contribute equally. Weiran Xu is the corresponding author.
well during realistic dialog scenario.

In this work, we proposed a simple but efficient framework named Domain-slot Related Information Awareness method (DRIA) based on the domain-slot related keywords extraction module and a random sampling strategy. Specifically, for the extraction module, we first use TF-IDF algorithm (Salton and Buckley, 1988) and CoreNLP (Manning et al., 2014) POS-tagging tool to extract several verbs and nouns in the dialogue as keywords, then we integrate them into our description. We believe that nouns and verbs always imply the topic or domain-slot related information in the dialogue. As shown in figure 1, the user query is "I want to book at allenbel for 4 nights for 8 people.". One of the user’s intention is to tell the agent how long to stay in the hotel, which can be expressed as “book hotel for 4 nights”. The verb "book" and noun "nights" imply the domain-slot: hotel-book stay. Similarly, "book", "nights" and "people" also imply the domain-slot: hotel-book people. Further, the random sampling strategy is designed for solving the problem of simplistic description. We extract keywords by random sampling during the training to ensure that the description of every domain-slot will become more informative each turn. During the evaluation, we inject all the extracted keywords into the description to provide the model with as much information as possible.

Our contributions are summarized as follows: (1) We propose an effective framework to construct domain-slot related keywords descriptions. To the best of our knowledge, we are the first to incorporate keywords information into the DST task. (2) We design a random sampling training strategy to integrate rich domain-slot related information during the training, which aims to improve generalization ability. (3) Experimental results show that our method outperforms most of the previous methods in the cross-domain few-shot DST settings, especially in the slot accuracy.

2 Methodology

2.1 Keyword-description

As shown in the figure 2, we built a keyword list for each slot. These keywords are mainly divided into two categories: domain-slot related verbs and nouns. Further, we add slot type (Lin et al., 2021b) into the keyword list which can prompt the output of the model according to the type of slot value and makes the model output more uniform for the same domain-slot. The format of the our description is "The [slot] of the [domain] which may include {slot_v} or {slot_n}, and its output type is [output_type]."

2.2 Keyword extraction module

The procedure of this extraction module is divided into three steps: (1) We traverse the entire dataset. For each domain-slot, if the domain-slot is mentioned in a turn of dialogue, mark each token’s part of speech in this turn (CoreNLP (Manning et al., 2014) POS-tagging tool is used here), then record each word in a list of this turn. (2) Count the word frequency of the list for each slot, and take the top 20. (3) Use TF-IDF algorithm (Salton and Buckley,
1988), we calculate the weight of words in the list corresponding to each slot, and then take the top 5 as the content of each category of keywords.

2.3 Random sampling strategy
In order to improve the domain generalization ability of the model, we propose a random sampling strategy which enriches the content of description during training. As shown in the figure 2, during training, we disarrange the list of keywords of each category, and then randomly select some keywords (or empty list) to build the description. In this way, for the same domain-slot, the input of description content may have a great probability of difference each time, which makes model understand the description content better, thus increasing the generalization ability of the model. During evaluating and testing, we do not use this random training method, but input all keywords into the model to provide as much information as possible.

2.4 Keywords-prompt DST
In this section, we define the dialogue history $C_t$ which is the accumulation of dialogues from the beginning to the current turn $t$. Each turn of dialogue is composed of the system and the user’s utterance. We record the dialogue history as $C_t = \{M_1, N_1, ..., M_n, N_n\}$, where $t$ stands for the conversation turn, $M$ and $N$ denotes the system and user, respectively. The i-th input of the model is composed of the dialogue history and the description of the i-th domain-slot:

$$\text{input}_i = C_t \ [\text{sep}] \ \text{Description}_i$$

where $[\text{sep}]$ indicates connector. The i-th output is the value of the i-th domain-slot corresponding to the description in the conversation status in the turn T. If there is no slot value in the conversation turn, the output is "None":

$$\text{output}_i = \text{model}(\text{input}_i)$$

Finally, we use cross entropy as loss function.

2.5 Training and evaluation process
First, we utilize the extraction module to get the keywords list of each domain-slot. During the training process, the keywords are extracted to build the description according to the random sampling strategy. In each turn, we traverse the description of each slot and connect the description and the context as the input. Then the model outputs the corresponding results. During the evaluation stage, the extracted keywords will be used to build the description, and the other steps are roughly the same as those in training stage. Note that during the few-shot domain fine-tuning, we randomly select (1%, 5%, 10%) of dataset for keyword extraction, and then use the same data for training. We use T5-small (Raffel et al., 2019) as our experimental model to align with T5-DST (Lin et al., 2021b).

3 Experiments

3.1 Dataset, metric and Evaluation
MultiWOZ 2.0 dataset (Budzianowski et al., 2018) provides turn-level annotations of dialogue states in 7 different domains. We evaluate our method on this dataset and follow the pre-processing and evaluation setup from (Wu et al., 2019), where restaurant, train, attraction, hotel, and taxi domains are used for training and testing. We use Joint Goal Accuracy that is the average accuracy of predicting all slot assignments for a given service in a turn correctly to evaluate the main results of models.

3.2 Baselines
(1) TRADE: Transferable dialogue state generator (Wu et al., 2019) which utilizes copy mechanism to facilitate domain knowledge transfer. (2) DSTQA: Dialogue state tracking via question answering over ontology graph (Zhou and Small, 2019). (3) T5DST: (Lin et al., 2021b) A slot description enhanced approach for zero-shot & few-shot cross-domain DST based on T5.

3.3 Implementation
To ensure model consistency with T5DST (Lin et al., 2021b), we implement DRIA based on the T5-small (60M parameters) model which has 6 encoder-decoder layers and the hidden size is 512. All models are trained using an AdamW optimizer with a base learning rate of 0.0001. For our few-shot cross-domain experiments, the models are first trained on 4 domains with batch size 8 for 2 epochs then fine-tuned with 1%, 5% and 10% of target domain data for 5 epochs respectively. We use 1 NVIDIA 3090 GPU for all of our experiments. Joint goal accuracy is used to evaluate the performance of the models. Predicted dialogue states are correct only when all of the predicted values exactly match the correct values.
<table>
<thead>
<tr>
<th>Methods</th>
<th>Attraction</th>
<th>Hotel</th>
<th>Restaurant</th>
<th>Taxi</th>
<th>Train</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baselines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRADE</td>
<td>35.88</td>
<td>57.55</td>
<td>63.12</td>
<td>19.73</td>
<td>37.45</td>
</tr>
<tr>
<td>DSTQA</td>
<td>N/A</td>
<td>70.47</td>
<td>71.60</td>
<td>N/A</td>
<td>50.18</td>
</tr>
<tr>
<td>T5DST w/ Naive</td>
<td>57.45</td>
<td>64.97</td>
<td>67.76</td>
<td>43.90</td>
<td>50.17</td>
</tr>
<tr>
<td>T5DST w/ Slot Type</td>
<td>58.57</td>
<td>66.05</td>
<td>69.89</td>
<td>44.96</td>
<td>50.63</td>
</tr>
<tr>
<td>Our method</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DRIA</td>
<td>66.46</td>
<td>69.13</td>
<td>73.83</td>
<td>48.84</td>
<td>56.64</td>
</tr>
<tr>
<td>- Slot n</td>
<td>64.27</td>
<td>67.79</td>
<td>70.65</td>
<td>46.74</td>
<td>52.10</td>
</tr>
<tr>
<td>- Slot v</td>
<td>65.90</td>
<td>68.53</td>
<td>71.85</td>
<td>48.09</td>
<td>55.88</td>
</tr>
<tr>
<td>- Slot v &amp; n</td>
<td>63.23</td>
<td>67.02</td>
<td>68.82</td>
<td>45.96</td>
<td>50.07</td>
</tr>
<tr>
<td>- Random sampling</td>
<td>63.97</td>
<td>67.86</td>
<td>71.23</td>
<td>46.37</td>
<td>53.55</td>
</tr>
</tbody>
</table>

Table 1: Few-shot experimental results based on JGA in MultiWOZ 2.0. We evaluate our proposed model with 1%, 5%, and 10% in-domain data, against TRADE (Wu et al., 2019) and DSTQA (Zhou and Small, 2019).

### 3.4 Prompt Description Variants

(1) Naive: Simple transformation of the slot name from "domain-slot" pair to "[slot] of the [domain]".

(2) Slot Type: A template for each slot type that follows "[slot type] of [slot] of the [domain]" to facilitates the knowledge transfer among different slots.

(3) Slot related verbs & noun: The format of the description is "The [domain] of the [slot] which may include {slot_v} or {slot_n}, and its output type is [output_type]." Note that "[output_type]" follows the format of "slot type" (Lin et al., 2021b).

### 3.5 Main Results

Table 1 shows the few-shot result on MultiWOZ 2.0, where all the methods are trained on four source domain then finetuned with the target domain data. We experiment with 1%, 5% and 10% of the target domain data. The results show that DRIA outperforms all baseline methods in all 5 domains under different data ratio settings. The margin is especially obvious under the condition of 1% in-domain data settings (e.g., 7.89% in the attraction compared with T5DST w/ Slot Type). This dramatic improvement can be attributed to the first introducing of keywords that bridges the gap between the source and target data distributions.

**Ablation Studies.** We conduct ablation study to better prove the effectiveness of keywords. As shown in Table 1, the model performance degrades whether slot verb or noun is discarded. When we completely abandon the keywords part, the model performance drops the most. Model performance also degrades in ablation study on random sampling strategy. However, as the proportion of target domain data increases, the impact of the lack of random sampling will decrease. We believe this is due to the gradual adaptation of the model to the dataset.

**Slot Accuracy Analysis.** Figure 3 show the slot accuracy of models using T5DST Slot Type and DRIA description. It can be seen that our slot related verbs & nouns description achieves better results on all 5 domain compared to the T5 slot type description, which further proves the effectiveness of our method. We speculate that explicit information about the domain-slot related keyword is important in few-shot scenario when the model does not have enough labeled data to capture the semantics of the new slot.

**Case Studies.** To further illustrate the effectiveness of our framework, figure 4 shows two representative samples of model prediction. In the first case, T5DST missed the domain-slot: restaurant-food as while our method correctly identified this domain-slot pair. our analysis is that Keyword descriptions provide rich information for the model to predict the correct slots as many as possible. In the second case, both T5DST and DRIA without the random sampling strategy regarded the attraction-name as hotel-name while DRIA avoided the mistake. For
Figure 4: Two representative samples in the test set of MultiWOZ 2.0 where “DRIA(w/o RM)” denotes the DRIA method without the random sampling strategy.

this case, the random sampling strategy may alleviate the misleading information and lack of domain generalization caused by the simplistic description to predict wrong slots as few as possible.

4 Conclusion

In this paper, we propose a simple but effective framework to tackle the few-shot cross-domain DST challenge. Specifically, we propose DRIA based on T5. This framework incorporates the domain-slot related information into the description to help the model distinguish the domain-slot more clearly. Further, we propose a random sampling strategy which enriches the content of description during training to improve the domain generalization ability of the model. Results on MultiWOZ dataset show that our method outperforms most of the previous methods in the cross-domain few-shot DST settings.

Limitation

This work has two main limitations: (1) The keywords are obtained based on statistical methods. There will be some dialogues which contain a certain slot while the keyword corresponding to the slot does not exist. In this case, the extracted keyword may be counterproductive to the model. (2) The input length of the T5 model (Raffel et al., 2019) limits the performance of model that requires user inputting a whole dialogue context and predict the dialogue state from scratch. When the description contains too many keywords or the dialogue length is too long, it will face a input truncation problem.

References


