Leveraging Slot Descriptions for Zero-Shot Cross-Domain Dialogue State Tracking

Zhaojiang Lin[1]*, Bing Liu[2], Seungwhan Moon[2], Paul Crook[2], Zhenpeng Zhou[2],
Zhiguang Wang[2], Zhou Yu[3], Andrea Madotto[1]*, Eunjoon Cho[2], Rajen Subba[2]
[1]The Hong Kong University of Science and Technology

[2]Facebook

[3]Columbia University

zlinao@ust.hk, bingl@fb.com

Abstract

Zero-shot cross-domain dialogue state tracking (DST) enables us to handle task-oriented dialogue in unseen domains without the expense of collecting in-domain data. In this paper, we propose a slot description enhanced generative approach for zero-shot cross-domain DST. Specifically, our model first encodes dialogue context and slots with a pre-trained self-attentive encoder, and generates slot values in an auto-regressive manner. In addition, we incorporate *Slot Type Informed Descriptions* that capture the shared information across slots to facilitate cross-domain knowledge transfer. Experimental results on the MultiWOZ dataset show that our proposed method significantly improves existing state-of-the-art results in the zero-shot cross-domain setting.

1 Introduction

Task-oriented dialogue systems are designed to assist users in performing daily activities, such as restaurant booking, travel planning, and online shopping. These virtual assistants provide natural language interfaces to services and online APIs (Rastogi et al., 2020). Based on users' needs, these systems frequently require support for new domains. However, the current state-of-the-art systems require a substantial amount of in-domain data to properly model a new domain. The data-collection process is both expensive and time-consuming, and thus it is very important to study methods that can build robust and scalable dialogue systems using little to no in-domain data.

The dialogue state tracking (DST) is an essential component of task-oriented dialogue systems that tracks users' requirements over multi-turn conversations. A popular formulation of the dialogue state is in the form of a list of slot-value pairs. In DST, tracking unseen slots in a new domain, a.k.a. zero-shot domain adaptation, is a significant challenge,
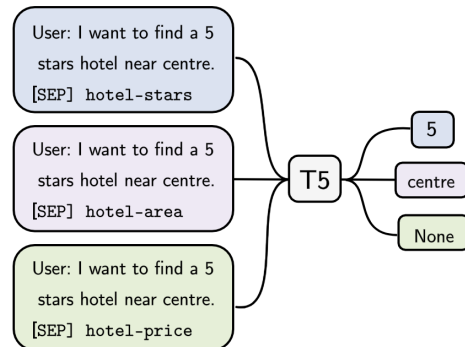

Figure 1: High-level description of the T5DST. The model takes dialogue history and slot name as input, and generates the value.

since the model has never seen in-domain training samples. There are two main lines of work to tackle this problem. The first proposes domain transferable models using copy mechanisms or ontology graph information (Wu et al., 2019; Zhou and Small, 2019). A limitation of such models is that they may not fully leverage pre-trained language models due to the specialized model architecture. The second line of work uses slot-descriptions as input to the model to facilitate the slot understanding (Rastogi et al., 2020). However, the provided slot descriptions are collected by crowd sourced human annotators and might be inconsistent among different domains. In general, the optimal approach for constructing slot descriptions in zero-shot settings remains unexplored.

In this work, we tackle the challenge of zero-shot cross-domain DST via leveraging large scale pre-trained sequence-to-sequence (seq2seq) models and with effective encoding of slot descriptions. We first introduce a generative DST model called T5DST, which models the relation of a slot and its dialogue context with a self-attentive encoder, and generates the slot value with a decoder in an autoregressive manner. This simple design allows us to effectively incorporate a pre-trained
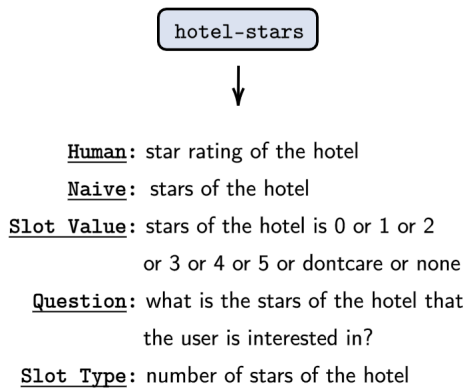

5640

```
                    ┌──────────────┐
                    │  hotel-stars │
                    └──────────────┘
                            │
                            ▼

         Human: star rating of the hotel
         Naive: stars of the hotel
    Slot Value: stars of the hotel is 0 or 1 or 2
                or 3 or 4 or 5 or dontcare or none
      Question: what is the stars of the hotel that
                the user is interested in?
     Slot Type: number of stars of the hotel
```

Figure 2: Slot description examples.

Slot Type	Slot Name
Number	hotel-book stay, hotel-book people, hotel-stars, train-book people, restaurant-book people
Location	train-destination, train-departure, taxi-destination, taxi-departure
Time	train-arriveby, train-leaveat, taxi-leaveat, restaurant-book time, taxi-arriveby
Boolean	hotel-parking, hotel-internet
Name	attraction-name, restaurant-name, hotel-name
Day	hotel-book day, train-day, restaurant-book day

Table 1: Slot type of slots in MultiWOZ. The full table is reported in Appendix A.1.

seq2seq model (e.g., T5 (Raffel et al., 2020)) without any task-specific modification. To further enhance the model's cross-domain transferability, we propose *Slot Type Informed Descriptions* that capture the shared information of different slots. Experimental results on the MultiWOZ benchmark (Budzianowski et al., 2018) suggest that 1) our model achieves significantly higher joint goal accuracy compared to existing results in zero-shot cross domain DST; 2) models using the proposed slot description formulation substantially outperform those using other slot description variants. Our contributions are summarized as the following:

- We propose a simple yet novel generative DST model based on T5 that significantly improves existing zero-shot cross-domain DST results;

- We investigate the effectiveness of different slot description formulations. To the best of our knowledge, this is the first work that comprehensively studies the effectiveness of slot descriptions in zero-shot cross-domain DST.

2   Related Work

Dialogue State Tracking has been of broad interest to the dialogue research community (Williams and Young, 2007; Williams et al., 2014; Heck et al., 2020; Liu et al., 2020; Wu et al., 2020; Madotto et al., 2020). Current state-of-the-art models (Chen et al., 2020; Lin et al., 2020; Heck et al., 2020; Hosseini-Asl et al., 2020; Ye et al., 2021; Li et al., 2020) trained with extensive annotated data have been shown promising performance in complex multi-domain conversations (Budzianowski et al., 2018). However, collecting large amounts of data for every

domain is costly and inefficient. To address this issue, several methods (Wu et al., 2019; Zhou and Small, 2019) have proposed for transferring prior knowledge of existing domains to new ones. On the other hand, Campagna et al. (2020) proposed an abstract dialogue model that leverages the ontology and in-domain templates to generate a large amount of synthesized data for domain adaptation. Different from their method, in this paper, we utilize a pre-trained seq2seq model and slot descriptions for cross-domain DST without any in-domain data.

Slot Description has been shown to be a promising technique in cross domain semantic parsing (Bapna et al., 2017; Shah et al., 2019; Namazifar et al., 2020). To encourage this line of research in DST as well, MultiWOZ2.1 (Eric et al., 2019) provides a further annotation for slot descriptions. Rastogi et al. (2020) incorporated slot descriptions for facilitating cross domain DST, while Gao et al. (2019, 2020) formulated DST as a question answering problem by casting a slot name into questions. However, these works did not show the effectiveness of slot descriptions, by comparing the performance of models with and without them. There is no study on how to construct slot descriptions. In this paper, we aim to fill this research gap by providing an empirical study on the different slot description formulations.

3   Methodology

3.1   T5DST

The design of our model follows the basis of generative question answering models. As illustrated in Figure 1, given a dialogue history which consists of an alternating set of utterances from two

Model	Joint Goal Accuracy					
	Attraction	Hotel	Restaurant	Taxi	Train	**Average**
TRADE	19.87	13.70	11.52	60.58	22.37	25.76
SUMBT*	22.60	**19.80**	16.50	59.50	22.50	28.18
SimpleTOD++	28.01±1.30	17.69±1.00	15.57±1.54	59.22±0.95	27.75±1.16	29.65±0.58
T5DST	**32.66**±0.10	18.73±1.67	**20.55**±0.96	**64.62**±0.24	**31.27**±0.47	**33.56**±0.54
w/ Human	31.92±1.42	20.72±0.35	20.09±0.67	64.12±0.28	28.83±1.28	33.14±0.17
w/ Naive	32.98±0.60	20.23±1.11	20.01±2.91	63.59±0.23	30.04±4.31	33.37±1.36
w/ Slot Value	32.86±0.56	20.03±0.87	16.65±0.37	**65.09**±0.12	29.66±2.75	32.86±0.48
w/ Question	32.45±0.39	19.79±1.18	**21.82**±0.91	64.40±0.27	32.61±1.38	34.21±0.63
w/ Slot Type	**33.09**±1.60	**21.21**±0.61	21.65±1.07	64.62±0.55	**35.42**±1.42	**35.20**±0.59

Table 2: Zero-shot cross-domain results in MultiWOZ 2.0. We run each experiment three times with different random seeds, and report the mean and standard deviation. Note that the reported averaged zero shot joint goal accuracy is not comparable to multi-domains joint goal accuracy. *Result from (Campagna et al., 2020).

speakers, denoted as $\mathcal{C}_t = \{U_1, R_1, \ldots, R_{t-1}, U_t\}$, we add the *"user:"* and *"system:"* prefixes to the user and system utterance respectively. Then all the utterances and slot names s_i are concatenated into a single sequence, i.e., *user:U_1 . . .system:R_{t-1} user:U_t [sep] s_i*. The sequence is used as the input to the encoder, and the decoder generates the corresponding slot value v_i:

$$v_i = Seq2seq(\mathcal{C}_t, s_i). \qquad (1)$$

The learning objective of this generation process is minimizing the negative log-likelihood of v_i given $\mathcal{C}_t$ and s_i, that is,

$$\mathcal{L} = -\sum_i^n \log p(v_i|\mathcal{C}_t, s_i), \qquad (2)$$

where n is the number of slots to be tracked.

We initialize the model parameters with T5 (Raffel et al., 2020), an encoder-decoder Transformer with relative position embeddings (Shaw et al., 2018) pre-trained on a massive amount of English text. We denote our model as *T5DST*. To incorporate slot descriptions into *T5DST*, we replace the slot name with its corresponding slot description as the model input.

3.2 Slot Type Informed Descriptions

Although different slots may have distinguishing names, they can share the same slot type. As shown in Table 1, the slot type of *hotel-stars* and *restaurant-book people* are both number slots, while *hotel-internet* and *hotel-parking* are both boolean slots. In light of these observations, we hypothesize that adding slot type information to the

slot description facilitates the knowledge transfer among different slots. We construct a template for each slot type that follows *"[slot type] of [slot] of the [domain]"*. We denote such a slot description as *Slot Type*. More details are available in Appendix A.1.

4 Experiments

4.1 Dataset and Evaluation

We evaluate the proposed method on the MultiWOZ 2.0 dataset (Budzianowski et al., 2018), which has 7 domains. We use the pre-processing and evaluation setup from Wu et al. (2019), where restaurant, train, attraction, hotel, and taxi domains are used for training, as the test set only contains these 5 domains.

In the zero-shot cross-domain experiments, the models are first trained with four domains and then evaluated on the test-set of the unseen domain. Joint goal accuracy is used to evaluate the performance of the models. The generated dialogue states are considered to be correct if and only if all of the predicted values exactly match the oracle values.

4.2 Implementation

We implement T5DST[1] based on the T5-small (60M parameters) model which has 6 encoder-decoder layers and the hidden size $d_{model} = 512$. All models are trained using an AdamW (Loshchilov and Hutter, 2018) optimizer with the initial learning rate of 0.0001. In all cross-domain zero-shot experiments, we train the models with batch size 128 for 5 epochs. For the few-shot

---

[1]Source code is available in https://github.com/facebookresearch/Zero-Shot-DST

Model	Attraction			Hotel			Restaurant			Taxi			Train		
	1%	5%	10%	1%	5%	10%	1%	5%	10%	1%	5%	10%	1%	5%	10%
TRADE	35.88	57.55	63.12	19.73	37.45	41.42	42.42	55.70	60.94	63.81	66.58	70.19	59.83	69.27	71.11
DSTQA	N/A	70.47	71.60	N/A	50.18	53.68	N/A	58.95	64.51	N/A	70.90	74.19	N/A	70.35	74.50
T5DST w/ Slot Type	58.77	65.72	69.54	43.07	50.71	54.86	57.63	61.86	63.47	70.12	73.67	74.70	70.82	74.18	77.57

Table 3: Few-shot experimental results 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).

experiments, the models are first trained on 4 domains for 5 epochs then fine-tuned with 1%, 5% and 10% of target domain data for 10 epochs. For full shot training, we train our model for at most 10 epochs with batch size 64 and early stop according to the loss in the validation set. Other hyperpramaters are same as zero-shot cross-domain setting. We use 8 NVIDIA V100 GPUs for all of our experiments. We use greedy decoding in test time.

4.3 Baselines

4.3.1 Models

TRADE. Transferable dialogue state generator (Wu et al., 2019) which utilizes copy mechanism to facilitate domain knowledge transfer.

SUMBT. Slot-utterance matching belief tracker (Lee et al., 2019) based on the language model BERT (Devlin et al., 2018).

DSTQA. Dialogue state tracking via question answering[2] over ontology graph (Zhou and Small, 2019).

SimpleTOD++. SimpleTOD (Hosseini-Asl et al., 2020) uses a single causal language model GPT2 (Radford et al., 2019) to generate the dialogue states. To adapt this model to a zero-shot cross-domain setting, we also provide the slot name as the model input. We denote this model as SimpleTOD++.

4.3.2 Slot Description Variants

Human. Human annotated slot descriptions collected in MultiWOZ2.1 (Eric et al., 2019) and used in MultiWOZ2.2 (Zang et al., 2020).

Naive. Simple transformation of the slot name from *"domain-slot"* to *"[slot] of the [domain]"*.

Slot Value. Following recent works (Zhang et al., 2019; Rastogi et al., 2020), slots are divided into

---

[2]We are aware of STARC (Gao et al., 2020). However, we are not able to compare to our results with their results because they use different training data.

categorical and *non-categorical* slots. For categorical slots, we incorporate the candidate values into the slot description, i.e., *"[slot] of the [domain] is [value-1] or [value-2]?"*. The order of values is random. For non-categorical slots, their descriptions are the same as aforementioned *Naive*.

Question. Similar to (Gao et al., 2019, 2020), we reformulate the slot into a natural language question, i.e., *"What is the [slot] of the [domain] that is the user interested in?"*.

4.4 Results & Discussion

4.4.1 Zero-Shot Cross-Domain

The results of the zero-shot cross domain experiments are shown in Table 2. Overall, T5DST achieves significantly higher performance in terms of averaged joint goal accuracy compared to the three baseline models TRADE, SUMBT, and SimpleTOD++. These results demonstrate that our model can effectively capture the slot-context relation, and thus generalize better in unseen domains.

Replacing slot-names with human annotated slot descriptions does not bring improvement to the zero-shot performance. This might because of the diverse and inconsistent human descriptions among different domains. For example, the human descriptions of *attraction-area* and *restaurant-area* are *"area to search for attractions"* and *"area or place of the restaurant"* respectively. Such inconsistent descriptions increase the challenge on slot understanding in the zero-shot learning setting. the model using naive slot descriptions gives similar performance to the one that uses original slot names. The two approaches lead to similar semantic representation of the slots. In contrast, incorporating slot values hurts the learning, leading to a lower joint goal accuracy in the restaurant domain. We observe that even though adding value candidates improve some of the categorical slots (e.g., restaurant-area 68.35% → 82.25% slot accuracy), it hurts the unseen non-categorical slots (e.g., restaurant-food 40.63% → 26.10% slot accuracy). These non-categorical slots are usually the
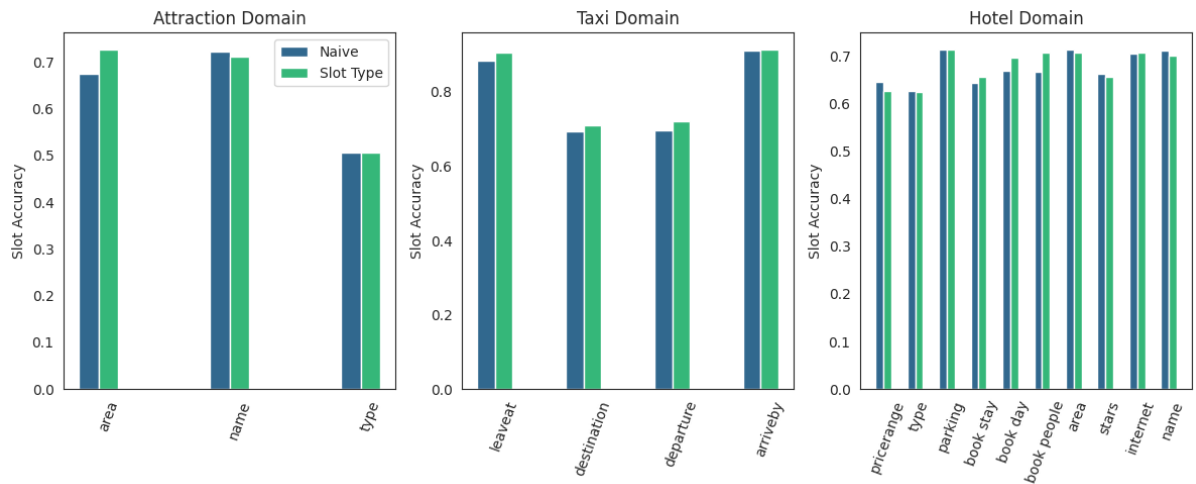
Figure 3: Slot accuracy in attraction, taxi, and hotel domains of MultiWOZ 2.0.
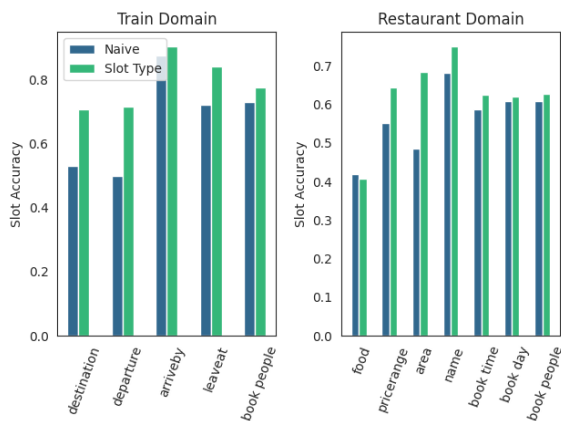

Figure 4: Slot accuracy in train and restaurant domains of MultiWOZ 2.0.

bottlenecks of joint goal accuracy. Finally, models trained with question style descriptions improves the performance in some domains, but fails in the others.

Our proposed slot type informed descriptions consistently improves the zero-shot performance of T5DST in all the domains. It produced an average of 2% joint goal accuracy improvement compared to human labeled and naive description formulations. This result indicates that slot type information may better capture the shared property (e.g., time, location) among different slots, thus facilitating the domain knowledge transferring for DST.

Figure 3 and 4 show the slot accuracy of models using *Naive* and *Slot Type* description. Compared to naive description, we obverse significant gain of time slots (e.g., arrive by and leave at), location slots (e.g., departure and destination), and number

slots (e.g., book stay and book people) by adding slot type information. We conjecture that explicit information about the target value (i.e., slot type) is important in the low resource condition when the model does not have enough data to capture the semantic meaning of a new slot.

4.4.2 Few-Shot Cross-Domain

We further conduct experiments in few-shot cross-domain settings, as in (Wu et al., 2019; Zhou and Small, 2019), where the models are first trained on 4 domains then fine-tuned with 1%, 5% and 10% of target domain data. As shown in Table 3, our model outperforms the DSTQA model in 4 out of 5 domains. Moreover, our approach is more practical in a real-world learning scenario as it does not require the supervision of a full ontology graph. We also conduct the full shot experiments and compare our model with previous methods. The reults are reported in Appendix A.2.

5 Conclusion

In this paper, we propose leveraging large scale pre-trained models with an effective slot description formulation to tackle the zero-shot cross-domain DST challenge. Specifically, we propose T5DST, a novel generative DST model based on the T5 language model, and incorporate Slot Type Informed Descriptions to facilitate cross-domain knowledge transfer. In the evaluation on the MultiWOZ dataset, our approach substantially improves existing results in both the zero-shot and few-shot settings.

References

Ankur Bapna, Gokhan Tur, Dilek Hakkani-Tur, and Larry Heck. 2017. Towards zero-shot frame semantic parsing for domain scaling. *arXiv preprint arXiv:1707.02363*.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026.

Giovanni Campagna, Agata Foryciarz, Mehrad Moradshahi, and Monica S Lam. 2020. Zero-shot transfer learning with synthesized data for multi-domain dialogue state tracking. *arXiv preprint arXiv:2005.00891*.

Lu Chen, Boer Lv, Chunxin Wang, Su Zhu, Bowen Tan, and Kai Yu. 2020. Schema-guided multi-domain dialogue state tracking with graph attention neural networks. In *AAAI 2020*.

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

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyag Gao, and Dilek Hakkani-Tur. 2019. Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*.

Shuyang Gao, Sanchit Agarwal, Tagyoung Chung, Di Jin, and Dilek Hakkani-Tur. 2020. From machine reading comprehension to dialogue state tracking: Bridging the gap. *arXiv preprint arXiv:2004.05827*.

Shuyang Gao, Abhishek Sethi, Sanchit Agarwal, Tagyoung Chung, and Dilek Hakkani-Tur. 2019. Dialog state tracking: A neural reading comprehension approach. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 264–273.

Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishauser, Hsien-Chin Lin, Marco Moresi, and Milica Gašić. 2020. Trippy: A triple copy strategy for value independent neural dialog state tracking. *arXiv preprint arXiv:2005.02877*.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. *arXiv preprint arXiv:2005.00796*.

Sungdong Kim, Sohee Yang, Gyuwan Kim, and Sang-Woo Lee. 2019. Efficient dialogue state tracking by selectively overwriting memory. *arXiv preprint arXiv:1911.03906*.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019. Sumbt: Slot-utterance matching for universal and scalable belief tracking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5478–5483.

Shiyang Li, Semih Yavuz, Kazuma Hashimoto, Jia Li, Tong Niu, Nazneen Rajani, Xifeng Yan, Yingbo Zhou, and Caiming Xiong. 2020. Coco: Controllable counterfactuals for evaluating dialogue state trackers. *arXiv preprint arXiv:2010.12850*.

Zhaojiang Lin, Andrea Madotto, Genta Indra Winata, and Pascale Fung. 2020. Mintl: Minimalist transfer learning for task-oriented dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3391–3405.

Zihan Liu, Genta Indra Winata, Zhaojiang Lin, Peng Xu, and Pascale Fung. 2020. Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8433–8440.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Andrea Madotto, Zihan Liu, Zhaojiang Lin, and Pascale Fung. 2020. Language models as few-shot learner for task-oriented dialogue systems. *arXiv preprint arXiv:2008.06239*.

Mahdi Namazifar, Alexandros Papangelis, Gokhan Tur, and Dilek Hakkani-Tür. 2020. Language model is all you need: Natural language understanding as question answering. *arXiv preprint arXiv:2011.03023*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696.

Darsh J Shah, Raghav Gupta, Amir A Fayazi, and Dilek Hakkani-Tur. 2019. Robust zero-shot cross-domain slot filling with example values. *arXiv preprint arXiv:1906.06870*.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468.

Jason D Williams, Matthew Henderson, Antoine Raux, Blaise Thomson, Alan Black, and Deepak Ramachandran. 2014. The dialog state tracking challenge series. *AI Magazine*, 35(4):121–124.

Jason D Williams and Steve Young. 2007. Partially observable markov decision processes for spoken dialog systems. *Computer Speech & Language*, 21(2):393–422.

Chien-Sheng Wu, Steven Hoi, Richard Socher, and Caiming Xiong. 2020. Tod-bert: Pre-trained natural language understanding for task-oriented dialogues. *arXiv preprint arXiv:2004.06871*.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019. Transferable multi-domain state generator for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 808–819.

Fanghua Ye, Jarana Manotumruksa, Qiang Zhang, Shenghui Li, and Emine Yilmaz. 2021. Slot self-attentive dialogue state tracking. *arXiv preprint arXiv:2101.09374*.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. Multiwoz 2.2: A dialogue dataset with additional annotation corrections and state tracking baselines. *arXiv preprint arXiv:2007.12720*.

Jian-Guo Zhang, Kazuma Hashimoto, Chien-Sheng Wu, Yao Wan, Philip S Yu, Richard Socher, and Caiming Xiong. 2019. Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking. *arXiv preprint arXiv:1910.03544*.

Li Zhou and Kevin Small. 2019. Multi-domain dialogue state tracking as dynamic knowledge graph enhanced question answering. *arXiv preprint arXiv:1911.06192*.

A Appendices

A.1 Slot Type Informed Description Construction

As shown in Table 4, each slot type has one prefix for appending to the beginning of the description. We used three different templates to construct the slot description. For all the booking slots (e.g., book people), we use *"[prefix] [slot] for the [domain] booking"*. For boolean slots, we use *"[prefix] [slot] in the [domain]"*. And for all the others, we use *"[prefix] [slot] of the [domain]"*. When a slot name (e.g., *train-day*) overlap with the slot type (e.g., *day*) or a slot does not fall into any slot type category (others), we simply set the prefix as an empty string.

A.2 Full Shot Results

To understand the full shot performance of our T5DST model and whether slot description is still helpful when there is enough training data, we also conduct the experiments in a full data setting. As shown in Table 5, using slot description only improves the joint goal accuracy by 0.56% in MultiWoz 2.0 and 0.30% in MultiWoz 2.1, which indicates that the description is less effective when there is a large amount of data for training.

Compared to prior models with zero-shot capability, T5DST shows promising performance. Compared to other state-of-the-art models that optimized for full shot training, our model achieve competitive results in MultiWoz 2.0, but inferior results on MultiWoz 2.1. We notice that there are many training strategies (e.g., token masking (Kim et al., 2019; Heck et al., 2020)), additional supervision (e.g., full ontology (Chen et al., 2020)), and label cleaning strategies (Heck et al., 2020)) that may impact final full-shot result. We also expect higher performance with a larger T5 model, such as T5-base or T5-large. However, achieving SOTA in full-scale training is out of the scope of this work.

Slot Type	Slot Name	Prefix	Examples
Number	hotel-book stay, hotel-book people, hotel-stars, train-book people, restaurant-book people	number of	number of people for the hotel booking
Location	train-destination, train-departure, taxi-destination, taxi-departure	location of	location of destination of the train
Time	train-arriveby, train-leaveat, taxi-leaveat, restaurant-book time, taxi-arriveby	time of	time of arrive by of the train
Boolean	hotel-parking, hotel-internet	whether have	whether have parking in the hotel
Name	attraction-name, restaurant-name, hotel-name	-	name of attraction
Day	hotel-book day, train-day, restaurant-book day	-	day for the hotel booking
Others	hotel-type, attraction-type, hotel-area, attraction-area, restaurant-food, restaurant-pricerange, restaurant-area	-	type of the hotel

Table 4: *Slot Type* description examples. We define one prefix for each slot type. The prefix is empty when a slot name overlap with the slot type or a slot does not fall into any slot type category (others).

Models	#Parameter	Zero-shot Inference	Joint Goal Accuracy	
			MWoz 2.0	MWoz 2.1
TRADE (Wu et al., 2019)	-	✓	48.62	45.6
STARC (Gao et al., 2020)	110M	✓	-	49.48
SUMBT (Lee et al., 2019)		✓	49.06	-
SGD-baseline (Rastogi et al., 2020)	110M	✓	-	43.4
T5DST	60M	✓	52.86	51.91
T5DST + Slot Type	60M	✓	53.42	52.21
DSTQA w/o span (Zhou and Small, 2019)	-	✗	51.44	51.17
MinTL (BART) (Lin et al., 2020)	400M	✗	52.10	53.67
SOM-DST (Kim et al., 2019)	340M	✗	52.32	53.68
SST (Chen et al., 2020)	110M	✗	51.17	55.23
TripPy (Heck et al., 2020)	110M	✗	-	55.29
SimpleTOD (Hosseini-Asl et al., 2020)	110M	✗	-	55.76

Table 5: Full shot results on MultiWOZ 2.0 and MultiWOZ 2.1.