Multi-Task Pre-Training for Plug-and-Play Task-Oriented Dialogue System

Anonymous ACL submission

Abstract

Pre-trained language models have been recently shown to benefit task-oriented dialogue (TOD) systems. Despite their success, existing methods often formulate this task as a cascaded generation problem which can lead to error accumulation across different sub-tasks and greater data annotation overhead. In this study, we present PPTOD, a unified plug-and-play model for task-oriented dialogue. In addition, we introduce a new dialogue multi-task pre-training strategy that allows the model to learn the primary TOD task completion skills from heterogeneous dialog corpora. We extensively test our model on three benchmark TOD tasks, including end-to-end dialogue modelling, dialogue state tracking, and intent classification. Experimental results show that PPTOD achieves new state of the art on all evaluated tasks in both high-resource and low-resource scenarios. Furthermore, comparisons against previous SOTA methods show that the responses generated by PPTOD are more factually correct and semantically coherent as judged by human annotators. [1]

1 Introduction

Task-oriented dialogue is often decomposed into three sub-tasks: (1) dialogue state tracking (DST) for tracking user's belief state; (2) dialogue policy learning (POL) for deciding which system action to take; (3) natural language generation (NLG) for generating dialogue response (Young et al., 2013).

Traditional approaches (Smith and Hipp, 1995; Young et al., 2013) adopt a modularized pipeline that addresses different sub-tasks with distinct dedicated modules. In contrast, recent systems (Wen et al., 2017; Eric et al., 2017; Lei et al., 2018; Shu et al., 2019) integrate all functionalities required to hold a dialogue into neural network models. With the advances in pre-trained language models (PLMs) (Radford et al., 2019; Devlin et al., 2019; Raffel et al., 2020), different systems based on PLMs have been proposed (Hosseini-Asl et al., 2020; Lin et al., 2020; Peng et al., 2021; Liu et al., 2021). Despite their differences, most existing methods formulate task-oriented dialogue as a cascaded generation problem, that is, the model can only solve latter sub-tasks by conditioning on the outputs of previous ones. For instance, to generate the response (NLG), the model must rely on the outputs of previous sub-tasks (i.e., DST and POL).

While impressive results are reported (Hosseini-Asl et al., 2020; Peng et al., 2021), we identify three major limitations in the cascaded formulation of their system design. (1) Firstly, as the model solves all sub-tasks in a sequential order, the errors accumulated from previous steps are propagated to latter steps (Li et al., 2017; Liu and Lane, 2018). (2) Secondly, the training data must be annotated for all sub-tasks. Such annotation requirement significantly increases the data curation overhead. More importantly, it precludes the model from using the large amount of existing data that is partially annotated (e.g., data only annotated with DST or NLG). (3) Thirdly, the results of different sub-tasks must be generated in a cascaded order which inevitably increases the system inference latency.

In this study, we propose a novel **P**lug-and-**P**lay **T**ask-**O**riented **D**ialogue (PPTOD) system. Figure 1 depicts an illustration of our approach. As seen, we integrate different dialogue modules (e.g. DST, POL, and NLG) into a unified model. Motivated by the concept of *in-context learning* (Brown et al., 2020), to steer the model to solve different TOD sub-task, we plug a task-specific natural language instruction, termed as *prompt*, into the dialogue context as the model input. This way, the generations of different sub-tasks are decoupled, leading to a greater flexibility of the model that brings us at least two advantages: (1) As different sub-tasks are solved separately, the model can learn from data that is partially annotated for different sub-tasks

[1] All code and models will be released upon publication.

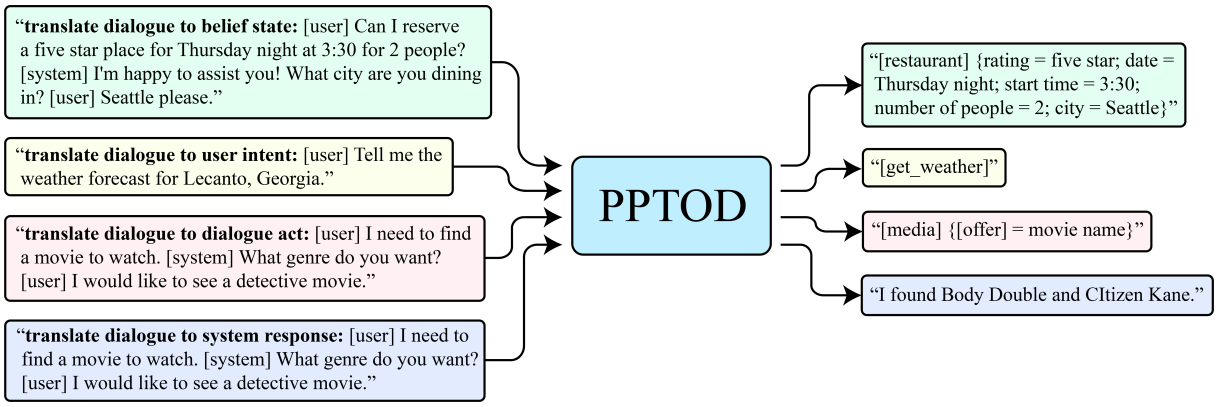

Figure 1: **Overview**: In the dialogue multi-task pre-training stage, we pre-train our model with four TOD-related tasks, including natural language understanding (NLU), dialogue state tracking (DST), dialogue policy learning (POL), and natural language generation (NLG). For each task, the model takes the dialogue context and the task-specific prompt as input and learns to generate the corresponding target text. Our learning framework allows us to train the model with partially annotated data across a diverse set of tasks. (best viewed in color)

(e.g., DST and NLG). (2) The outputs of different sub-tasks are generated in parallel which alleviates the problem of error accumulation and reduces the system inference latency.

Inspired by recent success of dialogue language model pre-training (Zhang et al., 2020c; Wu et al., 2020; Peng et al., 2021), we propose a dialogue multi-task pre-training strategy that equips our model with the primary TOD task completion skills. Specifically, initialized with T5 (Raffel et al., 2020), we pre-train our model on a heterogeneous set of dialog corpora that consist of partially-annotated data. To build the pre-training corpora, we collect and combine eleven human-written multi-turn dialogue corpora. The collected datasets are partially annotated for some of the TOD-related tasks, including natural language understanding (NLU), dialogue state tracking (DST), dialogue policy learning (POL), and natural language generation (NLG). In total, the pre-training corpora contain over 2.3M utterances across over 80 domains (see more details in Table 1). When applying the pre-trained PPTOD to a new task, we fine-tune it using the same learning objective as in the pre-training stage.

We evaluate PPTOD on a wide range of benchmark TOD tasks, including end-to-end dialogue modelling, dialogue state tracking, and intent classification. Comparisons against previous state-of-the-art approaches show that PPTOD achieves better performance in both full-training and low-resource settings as judged by automatic and human evaluations. In summary, our contributions are:

- A novel model, PPTOD, that effectively leverages pre-trained language models for task-oriented dialogue tasks.
- A new dialogue multi-task pre-training strategy that augments the model's ability with heterogeneous dialogue corpora.
- Extensive evaluations on three benchmark TOD tasks reporting state-of-the-art results in both full-training and low-resource settings.
- In-depth analysis that further reveals the merits of our model design and the proposed multi-task pre-training strategy.

2 Related Work

Task-Oriented Dialogue. Task-oriented dialogue aims at accomplishing user's goal. Traditional systems (Williams and Young, 2007; Young et al., 2013) adopt a pipelined approach that requires dialogue state tracking for understanding user's goal, dialogue policy learning for deciding which system action to take, and natural language generation for generating dialogue responses.

Recently, to simplify the modelling effort, researchers have shifted their attention to building neural network models that address the TOD sub-tasks (Wen et al., 2017; Eric et al., 2017; Lei et al., 2018; Liang et al., 2020). With the advances in pre-trained language models (PLMs), Budzianowski and Vulić (2019) first applied the GPT-2 model for the NLG task. Lin et al. (2020) and Yang et al. (2021) moved one step forward and utilized pre-trained language models to solve all TOD sub-tasks conditioned on the history of oracle belief states. Based on the GPT-2 model, Hosseini-Asl et al.

(2020) proposed a cascaded model, SimpleTOD, that addresses all TOD sub-tasks without using the oracle information. To improve the system performance, Peng et al. (2021) and Liu et al. (2021) applied dialogue pre-training over external dialogue corpora. However, both methods require the pre-training data to be fully annotated for all TOD sub-tasks (i.e., DST, POL, and NLG) which greatly limits the amount of data they can use. Additionally, Liu et al. (2021) achieved better results with noisy chanel model that requires two additional language models for outputs re-scoring. Unlike their approach, we address the task of task-oriented dialogue with a single unified model.

Language Model Pre-training. The research community has witnessed remarkable progress of pre-training methods in a wide range of NLP tasks, including language understanding (Peters et al., 2018; Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019) and text generation (Radford et al., 2019; Lewis et al., 2020; Raffel et al., 2020).

In the dialogue domain, many models are pre-trained on open-domain conversational data like Reddit. Based on GPT-2, Transfertransfo (Wolf et al., 2019b) achieves good results on ConvAI-2 competition. As another extension of GPT-2, DialoGPT (Zhang et al., 2020c) performs well in generating open-domain dialogue response. ConveRT (Henderson et al., 2020) is a language model with dual-encoder built for the task of response selection. PLATO (Bao et al., 2020) pre-trains a model with discrete latent variable structure for the response generation task. Wu et al. (2020) adapts BERT with TOD pre-training and achieves strong performances on four dialogue understanding tasks.

Pre-training on Supplementary Data. Recent work (Phang et al., 2018; Aghajanyan et al., 2021) found that supplementary training on the tasks with intermediate-labelled data improves the performance of the fine-tuned models on GLUE natural language understanding benchmark (Wang et al., 2018). Our work studies a similar supplementary training setup with intermediate-labelled data for task-oriented dialogue systems. Unlike previous work, we use a single multi-task model for all relevant sub-tasks in task-oriented dialogue systems.

3 Methodology

In this section, we first discuss the datasets and learning objective used in the proposed dialogue multi-task pre-training. Then we introduce how to apply the pre-trained PPTOD for a new task.

Dataset	Data Annotation				Utter.	Dom.
	NLU	DST	POL	NLG		
MetaLWOZ	×	×	×	✓	822,932	47
SNIPS	✓	×	×	×	25,682	9
CLINC	✓	×	×	×	45,000	10
ATIS	✓	×	×	×	10,772	1
KVRET	×	✓	×	✓	31,504	3
WOZ	×	✓	×	✓	15,248	1
CamRest676	×	✓	×	✓	10,976	1
MSR-E2E	×	✓	✓	✓	72,238	3
Frames	×	✓	✓	✓	38,316	1
TaskMaster	×	✓	✓	✓	540,688	6
Schema-Guided	×	✓	✓	✓	757,380	17

Table 1: The summary of data annotations and number of utterances (Utter.) as well as domains (Dom.) for all pre-training corpora. All datasets are partially annotated for some of the TOD-related tasks, including natural language understanding (NLU), dialogue state tracking (DST), dialogue policy learning (POL), and natural language generation (NLG).

3.1 Pre-training Datasets

To construct the pre-training corpus, we collect eleven human-written multi-turn task-oriented dialogue corpora, including MetaLWOZ (Lee et al., 2019b), SNIPS (Coucke et al., 2018), CLINC (Larson et al., 2019), ATIS (Amin, 2019), KVRET (Eric et al., 2017), WOZ (Mrkšić et al., 2017), MSR-E2E (Li et al., 2018), Frames (El Asri et al., 2017), TaskMaster (Byrne et al., 2019), and Schema-Guided (Rastogi et al., 2020). In total, there are over 2.3M utterances across 80 domains. In Table 1, we provide the details of data annotations and utterance/domain statistics of all datasets.[2]

3.2 Dialogue Multi-Task Pre-training

Motivated by previous work (McCann et al., 2018; Keskar et al., 2019; Raffel et al., 2020) that unify multiple NLP tasks into a common format, we cast all TOD-related tasks that we consider into the same plug-and-play text generation problem. To specify the target task, we plug a task-specific prompt into the dialogue context as the model input. Figure 1 depicts an illustration of our approach.

In the multi-task pre-training stage, each training sample is represented as:

$$d = (z_t, x, y), \quad (1)$$

where t denotes the TOD task that the sample d belongs to, and $t \in \{\text{NLU}, \text{DST}, \text{POL}, \text{NLG}\}$. z_t is

[2]More dataset descriptions are provided in Appendix A.

Algorithm 1: Dialogue Multi-Task Pre-Training

Input : Dataset $\mathcal{D} = \{(z_t, x, y)_i\}_{i=1}^{|\mathcal{D}|}$; model trainer $\mathcal{T}$ that takes batches of training data as input to optimize the model parameters Θ; maximum number of epochs e_{max};

for *epoch* $e = 1, ..., e_{\text{max}}$ **do**
  Shuffle $\mathcal{D}$ by mixing data from different tasks;
  for B *in* $\mathcal{D}$ **do**
    Invoke trainer $\mathcal{T}$, using one batch of training data $B = \{(z_t, x, y)_k\}_{k=1}^{|B|}$ as input to optimize the model using $\mathcal{L}_\Theta$ (Eq. (2)).
  end
end

Output : Trained Model Θ

the task-specific prompt of the form "`translate dialogue to A:`", with `A` corresponding to "user intent", "belief state", "dialogue act", and "system response" for the tasks of NLU, DST, POL, and NLG, respectively. x denotes the input dialogue context which is a concatenation of all previous utterances in the dialogue - both system's and user's. And y denotes the target output text.

As an example presented in Figure 1, to perform the user intent classification task (i.e., NLU), the model is fed with the sequence *"translate dialogue to user intent: [user] Tell me the weather forecast for Lecanto, Georgia."* and is trained to generate the user intent label text *"[get_weather]"*.

Learning. The model is trained with a maximum likelihood objective. Given the training sample $d = (z_t, x, y)$, the objective $\mathcal{L}_\Theta$ is defined as

$$\mathcal{L}_\Theta = -\sum_{i=1}^{|y|} \log P_\Theta(y_i | y_{<i}; z_t, x), \qquad (2)$$

where Θ is the model parameters.

In the multi-task pre-training stage, the model is trained to perform all TOD-related tasks with data annotated for different tasks. To optimize the model parameters Θ, we use mini-batch based optimization approach as shown in Algorithm 1.

3.3 Fine-Tuning to a New Task

When applying the pre-trained PPTOD to a new downstream task with task-specific labelled data, we use the same learning objective Eq. (2) as in the dialogue multi-task pre-training stage.

3.4 Implementation Details

In this work, we report results of PPTOD with three model sizes: $\text{PPTOD}_{\text{small}}$, $\text{PPTOD}_{\text{base}}$, and $\text{PPTOD}_{\text{large}}$. These three models are initialized with T5-small, T5-base, and T5-large models (Raffel et al., 2020) that contain ∼60M, ∼220M, and ∼770M parameters, respectively. We pre-train the model with different configurations on our collected pre-training corpora for 10 epochs. The training samples are truncated to ensure a maximal length of 1024. The models are trained using Adam optimizer (Kingma and Ba, 2015) with a learning rate of 5e-5 and a batch size of 128. Our implementation is based on the Huggingface Library (Wolf et al., 2019a).

4 Experiments

We test PPTOD on three benchmark TOD tasks: (1) end-to-end dialogue modelling; (2) dialogue state tracking; and (3) user intent classification.

4.1 End-to-End Dialogue Modelling

End-to-end dialogue modelling aims at evaluating the model in the most realistic, fully end-to-end setting, where the generated dialogue states are used for the database search and response generation (Zhang et al., 2020b; Hosseini-Asl et al., 2020).

4.1.1 Dataset and Evaluation Metric

We conduct experiments on the benchmark MultiWOZ 2.0 (Budzianowski et al., 2018) and 2.1 (Eric et al., 2020) datasets.[3] In MultiWOZ, the generation of response is not only related to the dialogue context, but also grounded on the database (DB) state. The DB state is automatically retrieved from a pre-defined database using the generated dialogue state (DST). Following previous studies, during inference, PPTOD first predicts the DST result to retrieve the DB state. Then, based on the retrieved DB state and the dialogue context, the results of POL and NLG are generated in parallel. In Section 5, we further compare the performance of our model with or without using the DB state as input.

For evaluation, we follow the original MultiWOZ guidance for all individual metrics: **Inform**, **Success**, and **BLEU** (Papineni et al., 2002). An overall measurement, i.e., combined score (Mehri et al., 2019), is also reported which is defined as **Combined** = (Inform + Success) × 0.5 + BLEU.

4.1.2 Baselines

We compare PPTOD with several strong baselines, including Sequicity (Lei et al., 2018), MD-Sequicity (Zhang et al., 2020b), DAMD (Zhang

[3]Note that, there is no overlap between the MultiWOZ dataset and our dialogue pre-training corpora.

Model	MultiWOZ 2.0				MultiWOZ 2.1			
	Inform	Success	BLEU	Combined Score	Inform	Success	BLEU	Combined Score
Sequicity	66.41	45.32	15.54	71.41	-	-	-	-
MD-Sequicity	75.72	58.32	15.40	82.40	-	-	-	-
DAMD	76.33	60.40	16.60	84.97				
MinTL†	84.88	74.91	17.89	97.78	-	-	-	-
HIER-Joint	80.50	71.70	19.74	95.84	-	-	-	-
SOLOIST	85.50	72.90	16.54	95.74	-	-	-	-
TOP§	85.20	72.90	17.00	96.05	-	-	-	-
TOP+NOD§	86.90	76.20	**20.58**	102.13	-	-	-	-
LABES-S2S	-	-	-	-	78.07	67.06	18.13	90.69
UBAR†, ‡	85.10	71.02	16.21	94.27	86.20	70.32	16.48	94.74
SimpleTOD	84.40	70.10	15.01	92.26	85.00	70.50	15.23	92.98
$\text{PPTOD}_{\text{small}}$	87.80	75.30	19.89	101.44	**88.89**	76.98	18.59	101.52
$\text{PPTOD}_{\text{base}}$	**89.20**	**79.40**	18.62	**102.92**	87.09	**79.08**	**19.17**	**102.26**
$\text{PPTOD}_{\text{large}}$	82.60	74.10	19.21	97.56	86.43	74.35	17.89	98.28

Table 2: End-to-end evaluation. †: the models require the history of oracle dialogue states when making predictions at current turn. ‡: UBAR scores are acquired with the author-released models. §: as the authors did not release their code, we cite the results of TOP and TOP+NOD on MultiWOZ 2.0 from the original paper (Liu et al., 2021).

Model	1% of training data				5% of training data				10% of training data				20% of training data			
	Inform	Succ.	BLEU	Comb.	Inform	Succ.	BLEU	Comb.	Inform	Succ.	BLEU	Comb.	Inform	Succ.	BLEU	Comb.
MD-Sequicity‡	-	-	-	-	49.40	19.70	10.30	44.85	58.10	34.70	11.40	57.80	64.40	42.10	13.00	66.25
DAMD†	34.40	9.10	8.10	29.85	52.50	31.80	11.60	53.75	55.30	30.30	13.00	55.80	62.60	44.10	14.90	68.25
SOLOIST†	58.40	35.30	10.58	57.43	69.30	52.30	11.80	72.60	69.90	51.90	14.60	75.50	74.00	60.10	15.25	82.29
MinTL‡	-	-	-	-	75.48	60.96	13.98	82.20	78.08	66.87	15.46	87.94	82.48	68.57	13.00	88.53
$\text{PPTOD}_{\text{small}}$	66.96	50.90	12.51	71.44	76.58	61.60	15.35	84.44	83.50	68.18	15.56	91.01	82.96	69.90	**17.02**	93.45
$\text{PPTOD}_{\text{base}}$	**74.42**	**52.44**	**12.99**	**76.41**	**79.86**	**63.48**	**14.89**	**86.55**	**84.42**	**68.36**	**15.57**	**91.96**	**84.94**	**71.70**	17.01	**95.32**
$\text{PPTOD}_{\text{large}}$	64.38	51.94	11.84	70.01	75.20	61.94	14.17	82.54	80.64	66.74	15.25	88.94	81.74	72.18	15.13	92.09

Table 3: Low-resource evaluation on MultiWOZ 2.0, where Succ. and Comb. denote the Success and Combined Score metrics, respectively. ‡ and † results are cited from Lin et al. (2020) and Peng et al. (2021).

et al., 2020b), MinTL (Lin et al., 2020), HIER-Joint (Santra et al., 2021), LABES-S2S (Zhang et al., 2020a), SimpleTOD (Hosseini-Asl et al., 2020), UBAR (Yang et al., 2021), and SOLOIST (Peng et al., 2021), TOP and TOP+Noisy Online Decoding (TOP+NOD) (Liu et al., 2021).

4.1.3 Full Training Evaluation

Table 2 shows the main results. On both MultiWOZ 2.0 and 2.1 datasets, PPTOD performs better than previous SOTA methods on seven out of eight metrics. In particular, it is worth mentioning that our model is a single architecture that does not require additional language models for re-ranking the outputs as in TOP+NOD (Liu et al., 2021).

4.1.4 Low-Resource Evaluation

To investigate the generalization ability of PPTOD, we evaluate it in a more challenging low-resource scenario. Following previous studies, we train our model on MultiWOZ 2.0 by varying the percentage of training data, ranging from 1% (∼80 samples) to 20% (∼1600 samples). We compare our model with several strong baselines, including MD-Sequicity, DAMD, SOLOIST, and MinTL.[4]

In each low-resource setting, we train our model five times with different random seeds and different selection of training data. The average scores over five runs are presented in Table 3.[5] As seen, PPTOD consistently outperforms all baseline models by a large margin. Notably, our performance gain is even larger when fewer samples are used for training. This indicates that PPTOD better leverages the prior knowledge from pre-training therefore achieving better results in the extreme low-resource settings. Furthermore, with 20% of training data, PPTOD can achieve results that are comparable to the scores of systems like SOLOIST that are trained with full dataset as reported in Table 2.

4.2 Dialogue State Tracking

Next, we evaluate PPTOD for the dialogue state tracking task. The experiments are conducted on the benchmark MultiWOZ 2.0 (Budzianowski et al., 2018) and 2.1 (Eric et al., 2020) datasets. For evaluation, the joint goal accuracy is reported.

[4]We did not compare results with TOP+NOD (Liu et al., 2021) since the authors did not release their code and models.

[5]Detailed numerical results can be found in Appendix B.

Model	MWOZ Joint Acc.(%)	
	2.0	2.1
Classification-based Approaches		
GLAD (Zhong et al., 2018)	35.57	-
GCE (Nouri and Hosseini-Asl, 2018)	36.27	-
FJST (Eric et al., 2020)	40.20	38.00
SUMBT (Lee et al., 2019a)	46.65	-
TOD-BERT (Wu et al., 2020)	-	48.00
DS-Picklist (Zhang et al., 2019) †	54.39	53.30
SST (Chen et al., 2020) †	51.17	55.23
TripPy (Heck et al., 2020)	-	55.29
CHAN (Shan et al., 2020) †	52.68	58.55
FPDSC-turn (Zhou et al., 2021) †	**55.03**	57.88
FPDSC-dual (Zhou et al., 2021) †	53.17	**59.07**
Generation-based Approaches		
Neural Reading (Gao et al., 2019)	41.10	-
TRADE (Wu et al., 2019)	48.62	46.00
COMER (Ren et al., 2019)	48.79	-
DSTQA (Zhou and Small, 2019) †	51.44	51.17
SOM-DST (Kim et al., 2020)	51.38	52.57
LABES-S2S (Zhang et al., 2020a)	-	51.45
MinTL (Lin et al., 2020)	52.10	53.62
SimpleTOD (Hosseini-Asl et al., 2020)	-	55.76
Seq2seq-DU (Feng et al., 2021)	-	56.10
UBAR (Yang et al., 2021)	52.59	56.20
SOLOIST (Peng et al., 2021)	53.20	56.85
$\text{PPTOD}_{\text{small}}$	51.50	56.47
$\text{PPTOD}_{\text{base}}$	53.37	57.10
$\text{PPTOD}_{\text{large}}$	**53.89**	**57.45**

Table 4: DST results. †: the models require a full pre-defined ontology for all possible domain-slot pairs.

4.2.1 Full Training Evaluation

We compare PPTOD with a wide range of existing methods that can be categorized into two classes: (1) classification-based approaches and (2) generation-based approaches. Table 4 shows the DST results. Compared to other generation-based approaches, $\text{PPTOD}_{\text{large}}$ obtains the highest accuracy on both datasets. The performance of our model is lower than the SOTA classification-based approaches. However, these methods operate on a fixed ontology and perform prediction over a pre-defined set of slot-value pairs (Zhang et al., 2019; Chen et al., 2020; Shan et al., 2020; Zhou et al., 2021). This idea of fixed ontology is not scalable, as in real world applications, the ontology is subject to constant change (Heck et al., 2020). In contrast, PPTOD directly generates the outputs, making it more adaptive and generalizable to new ontology labels in real world applications.

4.2.2 Low-Resource Evaluation

To investigate how well PPTOD performs with limited training samples on the downstream task, we evaluate it in a simulated low-resource setting. Specifically, we train the model on MultiWOZ 2.0 by varying the percentage of training data (i.e., 1%, 5%, 10%, and 20%). We compare PPTOD with three strong generation-based baselines, including SimpleTOD, MinTL, and SOLOIST, using the official code released by the authors.

Model	Training Size (%)			
	1	5	10	20
SimpleTOD	7.91±1.07	16.14±1.48	22.37±1.17	31.22±2.32
MinTL	9.25±2.33	21.28±1.94	30.32±2.14	35.96±1.25
SOLOIST	13.21±1.97	26.53±1.62	32.42±1.13	38.68±0.98
$\text{PPTOD}_{\text{small}}$	27.85±0.77	39.07±0.85	42.36±0.29	45.98±0.38
$\text{PPTOD}_{\text{base}}$	29.72±0.61	40.20±0.39	43.45±0.64	46.96±0.40
$\text{PPTOD}_{\text{large}}$	**31.46±0.41**	**43.61±0.42**	**45.96±0.66**	**48.95±0.13**

Table 5: Low-resource DST Evaluation: The means and standard deviations over five runs are reported.

Table 5 shows the experimental results. As seen, in all settings, PPTOD outperforms other baselines by a large margin. In the extreme scenario, with only 1% of training data, PPTOD surpasses the strongest SOLOIST model by 18 points of accuracy. This demonstrates that our model is more generalizable and can be better applied to new tasks where the amount of training data is limited.

4.3 Intent Classification

The goal of intent classification, i.e., NLU, is to classify the user's intent based on the user's utterance. We conduct experiments on the benchmark Banking77 dataset (Casanueva et al., 2020) that contains data with 77 different intents. Following previous studies (Casanueva et al., 2020; Peng et al., 2021), we test our model in both full training and low-resource settings. In the low-resource setting, we vary the number of training samples per intent from 10 to 30. The standard classification accuracy is reported for evaluation.

We compare PPTOD with several strong baselines, including BERT-Fixed, BERT-Tuned, USE+ConveRT (Casanueva et al., 2020), USE (Yang et al., 2020), ConveRT (Henderson et al., 2020), and SOLOIST (Peng et al., 2021). It is worth mentioning that all compared baselines are classification-based approach that uses classifier with a softmax layer to make the prediction over the pre-defined intent set. In contrast, as described in section §3.2, PPTOD solves the classification task as a generation problem by directly generating the text of intent label. Therefore, when adapting to a new classification task, PPTOD is more flexible and no extra model parameters are required.

In the experiments, we train PPTOD for five runs with different selection of training data and random

Model	Generation Mode	DB	End-to-End Dialogue Modelling				Inference Measurement	
			Inform↑	Success↑	BLEU↑	Combined Score↑	Latency (ms)↓	Speedup↑
SOLOIST	Cascaded	✓	85.50	72.90	16.54	95.74	208.69	1.00×
MinTL	Cascaded	✓	84.88	74.91	17.89	97.78	78.82	2.65×
T5-small	Cascaded	×	83.60	71.20	18.09	95.49	38.70	5.39×
		✓	84.10	73.70	18.03	96.93	39.78	5.25×
	Plug-and-Play	×	84.70	72.80	**18.52**	97.27	**14.17**	**14.73**×
		✓	**85.10**	**75.10**	17.82	**97.92**	19.52	10.69×

Table 6: Comparison between plug-and-play and cascaded generation. ↑: higher is better and ↓: lower is better.

Model	# of Training Samples		
	10	30	full
BERT-Fixed†	67.55	80.07	87.19
BERT-Tuned†	83.42	90.03	93.66
USE†	84.23	89.74	92.81
ConveRT†	83.32	89.37	93.01
USE+ConveRT†	**85.19**	90.57	93.36
SOLOIST‡	78.73	89.28	93.80
$\text{PPTOD}_{\text{small}}$	78.87±0.36	87.88±0.26	93.27±0.39
$\text{PPTOD}_{\text{base}}$	82.81±0.45	89.64±0.28	93.86±0.22
$\text{PPTOD}_{\text{large}}$	84.12±0.23	**90.64±0.29**	**94.08±0.15**

Table 7: Results on Banking77 dataset. † and ‡ are cited from Casanueva et al. (2020) and Peng et al. (2021).

seeds. The average scores and standard deviations are reported in Table 7. We see that PPTOD is comparable with existing methods. On low-resource-30 and full training settings, $\text{PPTOD}_{\text{large}}$ achieves the best results. Our performance gains are even more remarkable given that PPTOD requires no extra parameters when solving the classification task.

5 Further Analysis

In this section, we present further discussions and empirical analyses of the proposed model.

5.1 Plug-and-Play vs Cascaded Generation

First, we compare our plug-and-play generation with the cascaded generation that is adopted by most existing studies. To this end, we fine-tune a T5-small model (without dialogue multi-task pre-training) on MultiWOZ 2.0 by either using the plug-and-play or the cascaded formulation. Moreover, we also examine the effect of DB state on the model performance. Specifically, for the plug-and-play model, when utilizing DB state, it first predicts the dialogue state (DST) to retrieve the DB state from the pre-defined database. Then, based on the DB state and dialogue context, the output of POL and NLG are generated in parallel. When ignoring the DB state, the plug-and-play model generates DST, POL, and NLG results in a fully paralleled fashion.

For evaluation, we report the results on end-to-end dialogue modelling task. In addition, we report the average inference latency and relative speedup of each model.[6] We compare our ablated models with two strong baselines, SOLOIST and MinTL.[7]

Table 6 presents the results. As seen, the plug-and-play models yield better results than their cascaded counterparts. One reason is that, for cascaded models, the previously generated results are explicitly used as model input for latter sub-tasks, which leads to error accumulation. Moreover, we see that using DB state generally improves the model performance for both plug-and-play and cascaded models as it provides the model with more grounding information. Furthermore, with DB state, our plug-and-play model achieves better overall score than MinTL with an around 4× speedup. This suggests that the plug-and-play formulation benefits the model both in terms of the generation accuracy as well as the inference latency.

5.2 Multi-Task Pre-Training Investigation

Next, we provide further analyses on the dialogue multi-task pre-training strategy. To quantify the importance of different pre-training data, we pre-train the T5-small model using data that is annotated for individual TOD-related task (i.e., NLU, DST, POL, and NLG). After pre-training, we then evaluate the models on three downstream TOD tasks using MultiWOZ 2.0 and Banking77 datasets. For end-to-end dialogue modelling and dialogue state tracking, we test the model in both 1% and full training settings. For intent classification, we measure the accuracy of models trained with either 10 training samples per intent or full training samples.

Table 8 presents the results with the first row showing the performance of vanilla T5-small model. As seen, without any pre-training, the

[6]The latency of each model is measured on a single Nvidia V100 GPU with a batch size of 1.

[7]We did not include TOP+NOD (Liu et al., 2021) for comparison as the authors did not release their code.

Pre-training Data Annotation				End-to-End Dialogue Modelling						Dialogue State Tracking		Intent Classification	
NLU	DST	POL	NLG	1% training			full training			1% training	full training	10 samples	full training
				Inform	Success	BLEU	Inform	Success	BLEU	Accuracy	Accuracy	Accuracy	Accuracy
×	×	×	×	53.28	36.08	11.65	83.10	72.40	18.17	17.44	50.55	75.12	92.91
✓	×	×	×	58.58	40.48	11.02	85.20	73.50	16.96	18.47	50.71	78.21	**93.37**
×	✓	×	×	66.10	46.40	11.26	86.30	74.90	18.52	**27.91**	51.48	75.97	93.03
×	×	✓	×	60.60	48.20	11.88	84.40	74.60	18.55	19.32	50.82	75.37	92.95
×	×	×	✓	59.38	40.78	12.34	83.60	74.70	**19.97**	17.82	50.58	75.61	92.97
✓	✓	✓	✓	**66.96**	**50.90**	**12.51**	**87.80**	**75.30**	19.89	27.85	**51.50**	**78.87**	93.27

Table 8: Performance of models pre-trained on data with different annotations. In the low-resource setting of different tasks, the average scores over five runs are reported. The last row reports the results of $\text{PPTOD}_{\text{small}}$.

vanilla T5-small model performs poorly in the low-resource setting of all evaluated tasks. This suggests that the prior knowledge from pre-training is indispensable for the model to achieve strong performances in the low-resource scenarios.

Moreover, we see that pre-training with data annotated for individual TOD-related task helps the model to attain better result in the corresponding downstream task. For example, pre-training with DST data notably improves the model performance in the downstream DST task both in low-resource and full-training settings. Similarly, pre-training with NLG data helps the model to get better BLEU score in the end-to-end dialogue modelling task.

Lastly, we see that the $\text{PPTOD}_{\text{small}}$ model attains the best results on most of the evaluation metrics. This suggests that the pre-training data with different annotations are compatible with each other and the joint utilization of all pre-training data helps the model to achieve the best overall performance.

5.3 Human Evaluation

We also conduct a human evaluation with the help of graders proficient in English using an internal evaluation platform. For evaluation, we randomly selected 50 dialogue sessions from the test set of MultiWOZ 2.0 dataset. We compare the results generated by the $\text{PPTOD}_{\text{base}}$ model against the results from the SOLOIST model. All generated results, plus the reference, are evaluated by five graders on a 3-point Likert scale (0, 1, or 2) for each of the following features[8]:

- **Understanding**: Whether the system correctly understands the user's goal.
- **Truthfulness**: Whether the system's response is factually supported by the reference.[9]
- **Coherency**: Whether the system's response is semantically coherent with the context.

[8]More evaluation details are provided in the Appendix C.

[9]For this metric, we only evaluate the results of PPTOD and SOLOIST. By definition, the reference gets a score of 2.0.

	Understanding	Truthfulness	Coherency	Fluency
Agreement	0.641	0.598	0.668	0.806
Reference	1.92	2.00	1.93	1.98
SOLOIST	1.78	1.29	1.64	1.97
PPTOD	**1.86**	**1.51**	**1.83**	**1.99**

Table 9: Human Evaluation Results

- **Fluency**: Whether the system's response is grammatically fluent and easy to understand.

Table 9 lists the results, with the first row showing strong inter-annotator agreements as measured by Fleiss' kappa coefficient (Fleiss et al., 1971). Comparing with SOLOIST, our model achieves better scores on all metrics. Moreover, on the truthfulness and coherency metrics, our model significantly outperforms SOLOIST as judged by Sign Test (p-value < 0.05), suggesting that PPTOD generates more factually correct and semantically coherent responses. Finally, we note that on the fluency metric, both systems perform comparably with the reference (p-value > 0.4). This shows that the fluency of such systems is largely guaranteed by the prior syntactic knowledge from pre-trained language models, which suggests that future research should focus more on the other aspects of dialog systems.

6 Conclusion

In this paper, we propose PPTOD, a unified model that supports both task-oriented dialogue understanding and response generation in a plug-and-play manner. In addition, we introduce a new dialogue multi-task pre-training strategy to further augment our model's ability in completing TOD-related tasks. Extensive experiments and analysis are conducted on three benchmark TOD tasks in both high-resource and low-resource settings. The automatic and human evaluations demonstrate that PPTOD outperforms the current SOTA systems in terms of various evaluation metrics.

References

Armen Aghajanyan, Anchit Gupta, Akshat Shrivastava, Xilun Chen, Luke Zettlemoyer, and Sonal Gupta. 2021. Muppet: Massive multi-task representations with pre-finetuning. *CoRR*, abs/2101.11038.

Hassan Amin. 2019. Atis airline travel information system, version 1.

Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. 2020. PLATO: pre-trained dialogue generation model with discrete latent variable. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 85–96. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Paweł Budzianowski and Ivan Vulić. 2019. Hello, it's GPT-2 - how can I help you? towards the use of pretrained language models for task-oriented dialogue systems. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 15–22, Hong Kong. Association for Computational Linguistics.

Pawel 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, Brussels, Belgium, October 31 - November 4, 2018*, pages 5016–5026. Association for Computational Linguistics.

Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Ben Goodrich, Daniel Duckworth, Semih Yavuz, Amit Dubey, Kyu-Young Kim, and Andy Cedilnik. 2019. Taskmaster-1: Toward a realistic and diverse dialog dataset. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4515–4524. Association for Computational Linguistics.

Iñigo Casanueva, Tadas Temcinas, Daniela Gerz, Matthew Henderson, and Ivan Vulic. 2020. Efficient intent detection with dual sentence encoders. *CoRR*, abs/2003.04807.

Lu Chen, Boer Lv, Chi Wang, Su Zhu, Bowen Tan, and Kai Yu. 2020. Schema-guided multi-domain dialogue state tracking with graph attention neural networks. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7521–7528. AAAI Press.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *CoRR*, abs/1805.10190.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: a corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 207–219, Saarbrücken, Germany. Association for Computational Linguistics.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Kumar Goyal, Peter Ku, and Dilek Hakkani-Tür. 2020. Multiwoz 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 422–428. European Language Resources Association.

Mihail Eric, Lakshmi Krishnan, Francois Charette, and Christopher D. Manning. 2017. Key-value retrieval networks for task-oriented dialogue. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 37–49, Saarbrücken, Germany. Association for Computational Linguistics.

Yue Feng, Yang Wang, and Hang Li. 2021. A sequence-to-sequence approach to dialogue state tracking. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the*

11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021, pages 1714–1725. Association for Computational Linguistics.

J.L. Fleiss et al. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Shuyang Gao, Abhishek Sethi, Sanchit Agarwal, Tagyoung Chung, and Dilek Hakkani-Tür. 2019. Dialog state tracking: A neural reading comprehension approach. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue, SIGdial 2019, Stockholm, Sweden, September 11-13, 2019*, pages 264–273. Association for Computational Linguistics.

Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishauser, Hsien-Chin Lin, Marco Moresi, and Milica Gasic. 2020. Trippy: A triple copy strategy for value independent neural dialog state tracking. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGdial 2020, 1st virtual meeting, July 1-3, 2020*, pages 35–44. Association for Computational Linguistics.

Matthew Henderson, Iñigo Casanueva, Nikola Mrksic, Pei-Hao Su, Tsung-Hsien Wen, and Ivan Vulic. 2020. Convert: Efficient and accurate conversational representations from transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 2161–2174. Association for Computational Linguistics.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Unifying question answering and text classification via span extraction. *CoRR*, abs/1904.09286.

Seokhwan Kim, Michel Galley, R. Chulaka Gunasekara, Sungjin Lee, Adam Atkinson, Baolin Peng, Hannes Schulz, Jianfeng Gao, Jinchao Li, Mahmoud Adada, Minlie Huang, Luis A. Lastras, Jonathan K. Kummerfeld, Walter S. Lasecki, Chiori Hori, Anoop Cherian, Tim K. Marks, Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, and Raghav Gupta. 2019. The eighth dialog system technology challenge. *CoRR*, abs/1911.06394.

Sungdong Kim, Sohee Yang, Gyuwan Kim, and Sang-Woo Lee. 2020. Efficient dialogue state tracking by selectively overwriting memory. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 567–582. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316, Hong Kong, China. Association for Computational Linguistics.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019a. SUMBT: slot-utterance matching for universal and scalable belief tracking. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5478–5483. Association for Computational Linguistics.

Sungjin Lee, Hannes Schulz, Adam Atkinson, Jianfeng Gao, Kaheer Suleman, Layla El Asri, Mahmoud Adada, Minlie Huang, Shikhar Sharma, Wendy Tay, and Xiujun Li. 2019b. Multi-domain task-completion dialog challenge. In *Dialog System Technology Challenges 8*.

Wenqiang Lei, Xisen Jin, Min-Yen Kan, Zhaochun Ren, Xiangnan He, and Dawei Yin. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1437–1447, Melbourne, Australia. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Xiujun Li, Yun-Nung Chen, Lihong Li, Jianfeng Gao, and Asli Celikyilmaz. 2017. Investigation of language understanding impact for reinforcement learning based dialogue systems. *CoRR*, abs/1703.07055.

Xiujun Li, Sarah Panda, JJ (Jingjing) Liu, and Jianfeng Gao. 2018. Microsoft dialogue challenge: Building

end-to-end task-completion dialogue systems. In *SLT 2018*.

Weixin Liang, Youzhi Tian, Chengcai Chen, and Zhou Yu. 2020. MOSS: end-to-end dialog system framework with modular supervision. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8327–8335. AAAI Press.

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 2020, Online, November 16-20, 2020*, pages 3391–3405. Association for Computational Linguistics.

Bing Liu and Ian R. Lane. 2018. End-to-end learning of task-oriented dialogs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 2-4, 2018, Student Research Workshop*, pages 67–73. Association for Computational Linguistics.

Qi Liu, Lei Yu, Laura Rimell, and Phil Blunsom. 2021. Pretraining the noisy channel model for task-oriented dialogue. *Trans. Assoc. Comput. Linguistics*, 9:657–674.

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

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *CoRR*, abs/1806.08730.

Shikib Mehri, Tejas Srinivasan, and Maxine Eskénazi. 2019. Structured fusion networks for dialog. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue, SIGdial 2019, Stockholm, Sweden, September 11-13, 2019*, pages 165–177. Association for Computational Linguistics.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. 2017. Neural belief tracker: Data-driven dialogue state tracking. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1777–1788, Vancouver, Canada. Association for Computational Linguistics.

Elnaz Nouri and Ehsan Hosseini-Asl. 2018. Toward scalable neural dialogue state tracking model. *CoRR*, abs/1812.00899.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2021. Soloist: Building task bots at scale with transfer learning and machine teaching. In *Transactions of the Association for Computational Linguistics*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

Jason Phang, Thibault Févry, and Samuel R. Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *CoRR*, abs/1811.01088.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

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. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8689–8696. AAAI Press.

Liliang Ren, Jianmo Ni, and Julian J. McAuley. 2019. Scalable and accurate dialogue state tracking via hierarchical sequence generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 1876–1885. Association for Computational Linguistics.

Bishal Santra, Potnuru Anusha, and Pawan Goyal. 2021. Hierarchical transformer for task oriented dialog systems. In *Proceedings of the 2021 Conference*

of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, pages 5649–5658. Association for Computational Linguistics.

Yong Shan, Zekang Li, Jinchao Zhang, Fandong Meng, Yang Feng, Cheng Niu, and Jie Zhou. 2020. A contextual hierarchical attention network with adaptive objective for dialogue state tracking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6322–6333. Association for Computational Linguistics.

Lei Shu, Piero Molino, Mahdi Namazifar, Hu Xu, Bing Liu, Huaixiu Zheng, and Gökhan Tür. 2019. Flexibly-structured model for task-oriented dialogues. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue, SIGdial 2019, Stockholm, Sweden, September 11-13, 2019*, pages 178–187. Association for Computational Linguistics.

Ronnie W. Smith and D. Richard Hipp. 1995. *Spoken Natural Language Dialog Systems: A Practical Approach*. Oxford University Press, Inc., USA.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019a. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019b. Transfertransfo: A transfer learning approach for neural network based conversational agents. *CoRR*, abs/1901.08149.

Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. 2020. TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 917–929, Online. Association for Computational Linguistics.

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 Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 808–819. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernández Ábrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 87–94. Association for Computational Linguistics.

Yunyi Yang, Yunhao Li, and Xiaojun Quan. 2021. UBAR: towards fully end-to-end task-oriented dialog system with GPT-2. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 14230–14238. AAAI Press.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Steve J. Young, Milica Gasic, Blaise Thomson, and Jason D. Williams. 2013. Pomdp-based statistical spoken dialog systems: A review. *Proc. IEEE*, 101(5):1160–1179.

Jianguo 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. *CoRR*, abs/1910.03544.

Yichi Zhang, Zhijian Ou, Min Hu, and Junlan Feng. 2020a. A probabilistic end-to-end task-oriented dialog model with latent belief states towards semi-supervised learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9207–9219. Association for Computational Linguistics.

Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020b. Task-oriented dialog systems that consider multiple appropriate responses under the same context. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9604–9611. AAAI Press.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020c. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 270–278. Association for Computational Linguistics.

Victor Zhong, Caiming Xiong, and Richard Socher. 2018. Global-locally self-attentive encoder for dialogue state tracking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1458–1467. Association for Computational Linguistics.

Jingyao Zhou, Haipang Wu, Zehao Lin, Guodun Li, and Yin Zhang. 2021. Dialogue state tracking with multi-level fusion of predicted dialogue states and conversations. *CoRR*, abs/2107.05168.

Li Zhou and Kevin Small. 2019. Multi-domain dialogue state tracking as dynamic knowledge graph enhanced question answering. *CoRR*, abs/1911.06192.

A Dataset Details

We elaborate the details of the dialogue datasets contained in the pre-training dialogue corpora.

- **MetaLWOZ** (Lee et al., 2019b) is designed for improving models' ability in generating natural language responses in unseen domains. It contains annotations for natural language generation (NLG) spanning over 47 domains.
- **SNIPS** (Coucke et al., 2018) is designed to help developing models capable of understanding users' intent (i.e., natural language understanding (NLU)). Its data consists of users' utterances gathered by crowdsourcing with over 20 intent labels across 9 domains.
- **CLINC** (Larson et al., 2019) is built for improving model's ability in detecting out-of-scope users' intents. It contains data with NLU annotations for 150 intents across 10 different domains.
- **ATIS** (Amin, 2019) is used for building intent classification (NLU) model. It contains data with 22 user intents from the airline travel information domain.
- **KVRET** (Eric et al., 2017) is a in-car personal assistant dataset with dialogues from three domains: calendar scheduling, weather information retrieval, and point-of-interest navigation. It contains annotations for user belief state (DST) and system response (NLG).
- **WOZ** (Mrkšić et al., 2017) and **CamRest676** (Wen et al., 2017) are collected with Wizard-of-Oz procedure. They contains dialogues with DST and NLG annotations from the restaurant domain.
- **MSR-E2E** (Li et al., 2018) contains dialogues from three domains, including movie-ticket booking, restaurant reservation, and taxi booking. The data are annotated for three TOD-related tasks: DST, POL, and NLG.
- **Frames** (El Asri et al., 2017) contains dialogues from the trip booking domain. Its data are annotated for three TOD-related tasks, including DST, POL, and NLG.
- **TaskMaster** (Byrne et al., 2019) includes dialogues from six domains. Its data is collected with Wizard-of-Oz and self-dialogue approaches. The dataset is annotated with DST, POL, and NLG.
- **Schema-Guided** (Rastogi et al., 2020) is used for the DSTC8 (Kim et al., 2019) dialogue competition. It contains dialogues from 17 domains and it supports three TOD-related tasks, including DST, POL, and NLG.

B Low-Resource MultiWOZ Evaluation

In Table 10, we show the results of our model on MultiWOZ 2.0 under different low-resource settings. To get more confident results, for each setting, we train our model for five runs with different selection of training data and different random seeds. The complete results along with the mean and standard deviations are presented in Table 10.

C Human Evaluation Guidelines

Please evaluate the system's response with respect to the following features: (1) Understanding; (2) Truthfulness; (3) Coherency; and (4) Fluency. In the following, we provide some guidelines regarding how to judge the quality of the system's response in terms of different features.

C.1 Understanding

This metric measures whether the system's response shows that the system is able to understand the goal and intent of the user. The definition of different scores are:

- 2: The system completely understands the user's goal and intent.
- 1: The system partially understands the user's goal and intent.
- 0: The system does not understand the user's goal and intent at all.

C.2 Truthfulness

This metric measures whether the system's response is factually supported by the reference response. The definition of different scores are:

- 2: The facts in the system's response are all supported by or can be inferred from the reference response.
- 1: The facts in the system's response are partially supported by the reference response.
- 0: The system's response is contradicted to the facts contained in the reference response.

Model	1% of training data				5% of training data				10% of training data				20% of training data			
	Inform	Succ.	BLEU	Comb.	Inform	Succ.	BLEU	Comb.	Inform	Succ.	BLEU	Comb.	Inform	Succ.	BLEU	Comb.
$\text{PPTOD}_{\text{small}}$																
run-1	68.50	54.90	13.98	75.68	78.40	61.50	14.78	84.73	79.70	68.70	17.10	91.30	83.40	71.10	17.05	94.30
run-2	64.70	50.20	12.19	69.64	75.20	61.30	15.85	84.10	87.00	67.30	13.89	91.04	82.80	68.90	17.03	92.88
run-3	65.30	46.10	10.79	66.49	75.40	60.80	15.99	84.09	84.30	68.10	15.33	91.50	83.20	70.00	17.01	93.61
run-4	64.80	51.00	12.43	70.33	77.20	59.70	15.75	84.20	84.50	71.90	14.51	92.71	82.40	69.40	17.93	93.83
run-5	71.50	52.30	13.14	75.04	76.70	64.70	14.37	85.07	78.00	64.90	16.99	88.44	83.00	70.10	16.10	92.65
average	66.96	50.90	12.51	71.44	76.58	61.60	15.35	84.44	83.50	68.18	15.56	91.01	82.96	69.90	17.02	93.45
std	2.67	2.88	1.06	3.46	1.18	1.67	0.65	0.39	3.33	2.26	1.29	1.40	0.34	0.74	0.58	0.61
$\text{PPTOD}_{\text{base}}$																
run-1	74.20	55.40	13.08	77.88	80.50	66.10	15.58	88.88	85.10	67.50	16.02	92.32	84.90	72.50	17.16	95.86
run-2	71.20	51.10	13.32	74.47	81.50	63.10	14.32	86.62	84.60	69.00	15.06	91.86	84.00	72.50	16.46	94.71
run-3	76.20	49.70	12.39	75.34	77.50	61.70	14.98	84.58	84.10	69.20	15.49	92.14	85.50	69.60	17.76	95.31
run-4	75.80	52.40	13.21	77.30	79.70	62.30	15.13	86.10	84.40	68.30	15.17	91.52	84.20	70.70	16.88	94.33
run-5	74.70	53.60	12.97	77.05	80.10	64.20	14.44	86.59	83.90	67.80	16.12	91.96	86.10	73.20	16.78	96.43
average	74.42	52.44	12.99	76.41	79.86	63.48	14.89	86.55	84.42	68.36	15.57	91.96	84.94	71.70	17.01	95.32
std	1.76	1.97	0.32	1.29	1.32	1.55	0.46	1.38	0.42	0.66	0.43	0.27	0.79	1.34	0.44	0.75
$\text{PPTOD}_{\text{large}}$																
run-1	64.40	51.90	11.30	69.45	75.20	59.80	14.01	81.51	79.30	64.60	14.82	86.77	82.10	69.70	14.68	90.58
run-2	65.50	53.20	12.01	71.36	74.30	64.10	14.98	83.18	80.40	67.80	15.01	89.11	81.70	72.20	15.61	92.56
run-3	66.20	50.80	11.94	70.49	76.90	62.30	14.01	83.61	81.30	69.20	16.23	91.48	80.90	70.80	14.33	90.18
run-4	62.70	52.60	12.20	69.85	76.20	60.70	13.45	81.90	82.30	66.90	14.99	89.59	83.10	73.50	15.83	94.13
run-5	63.10	51.20	11.73	68.88	73.40	62.80	14.42	82.52	79.90	65.20	15.21	87.76	80.90	74.70	15.21	93.01
average	64.38	51.94	11.84	70.01	75.20	61.94	14.17	82.54	80.64	66.74	15.25	88.94	81.74	72.18	15.13	92.09
std	1.34	0.88	0.31	0.85	1.26	1.53	0.51	0.78	1.06	1.68	0.50	1.61	0.82	1.80	0.56	1.49

Table 10: Low-Resource Experiments on MultiWOZ: The average and std rows show the mean and standard deviation of results from five different runs. The Succ. and Comb. denote Success and Combined Score, respectively.

C.3 Coherency

This metric measures whether the system's response is logically coherent with the dialogue context. The definition of different scores are:

- 2: The system's response is logically coherent with the dialogue context.
- 1: The system's response contains minor information that is off the topic of the dialogue context.
- 0: The system's response is completely irrelevant to the dialogue context.

C.4 Fluency

The metrics measures the fluency of the system's response. The definition of different scores are:

- 2: The system's response is grammatically correct and easy to understand.
- 1: The system's response contains minor errors but they do not affect your understanding.
- 0: The system's response does not make sense and it is unreadable.

D Case Study

Table 11 presents a generated dialogue example from the $\text{PPTOD}_{\text{base}}$ model. The user starts the conversation by asking for an expensive restaurant that serves Indian food for dinner. PPTOD finds 14 restaurants that satisfy the user's goal and asks the user for a preferred location. We can see that, when the user states no preference on the restaurant location, PPTPD correctly updates the dialogue state by adding the area information which is missed by the oracle information. Then the user switches the dialogue topic for booking a hotel. Through the dialogue trajectory, we see that PPTOD completes the dialogue by successfully providing the user the necessary information such as number of hotel choices (at turn 3) and the booking reference number (at turn 6). When finding the user's booking request cannot be fulfilled (at turn 5), the models asks the user for an alternative option. Moreover, this example also demonstrates that PPTOD is able to deal with some NLU challenges displayed in the conversations. For example, at turn 4, the user already provides the information about the Gonville Hotel. But only after the user describes the intention of booking the hotel at turn 5, the model updates the name of hotel in the dialogue state based on the co-referenced information from the previous turn. Interestingly, the hotel name is ignored by the oracle dialogue state but our model correctly detects it. The dialogue understanding ability of PPTOD can also be observed in turn 6, in which it updates the hotel stay in the belief state from 2 days to 1 day after the user provides the corresponding information.

MUL0113		Dialogue
Turn 1	User Utterance	i am in need of an expensive indian restaurant for dinner .
	Generated Belief State	[restaurant] {food = indian, pricerange = expensive}
	Oracle Belief State	[restaurant] {food = indian, pricerange = expensive}
	Delexicalized Response	there are [value_choice] [value_price] [value_food] restaurants in cambridge . what area would you like to dine in ?
	Lexicalized Response	there are 14 expensive indian restaurants in cambridge . what area would you like to dine in ?
	Oracle Reference	there are 14 such restaurants . do you want to be in a specific part of town ?
Turn 2	User Utterance	i don't care what part of town it is in , i would like a phone number please .
	Generated Belief State	[restaurant] {food = indian, pricerange = expensive, area = don't care}
	Oracle Belief State	[restaurant] {food = indian, pricerange = expensive}
	Delexicalized Response	[value_name] is a [value_food] restaurant in the [value_area] part of town . their phone number is [value_phone] .
	Lexicalized Response	curry garden is a indian restaurant in the center part of town . their phone number is 01223302330 .
	Oracle Reference	sure , i am sorry . i have curry garden their phone number is 01223302330 . would you like me to reserve a table for you ?
Turn 3	User Utterance	not at this time . i am looking for a 3-star hotel , though . no guest house .
	Generated Belief State	[hotel] {stars = 3, type = hotel}; [restaurant] {food = indian, pricerange = expensive}
	Oracle Belief State	[hotel] {stars = 3, type = hotel}; [restaurant] {food = indian, pricerange = expensive}
	Delexicalized Response	i have [value_choice] [value_stars] star [value_type] in the [value_area] . [value_name] is a [value_stars] star [value_type] . would you like to book a room ?
	Lexicalized Response	i have 2 3-star hotel in the center . gonville hotel is a 3-star hotel . would you like to book a room ?
	Oracle Reference	i have found the gonville hotel . it is a 3 star hotel . would you like me to book it for you ?
Turn 4	User Utterance	not yet . is the gonville hotel in the same price range as the curry garden ?
	Generated Belief State	[hotel] {stars = 3, type = hotel}; [restaurant] {food = indian, pricerange = expensive}
	Oracle Belief State	[hotel] {stars = 3, type = hotel}; [restaurant] {food = indian, pricerange = expensive}
	Delexicalized Response	yes , it is in the [value_price] price range . would you like me to book it for you ?
	Lexicalized Response	yes , it is in the expensive price range . would you like me to book it for you ?
	Oracle Reference	yes , it is an expensive hotel .
Turn 5	User Utterance	can you book a room for 2 people for 2 nights starting on sunday ?
	Generated Belief State	[hotel] {stars = 3, type = hotel, name = gonville hotel, people = 2, day = sunday, stay = 2}; [restaurant] {food = indian, pricerange = expensive}
	Oracle Belief State	[hotel] {stars = 3, type = hotel, people = 2, day = sunday, stay = 2}; [restaurant] {food = indian, pricerange = expensive}
	Delexicalized Response	i am sorry , but the [value_name] is not available for that stay . would you like to try a shorter stay ?
	Lexicalized Response	i am sorry , but the gonville hotel is not available for that stay . would you like to try a shorter stay ?
	Oracle Reference	the booking was unsuccessful . would you like to try for another day or a shorter stay ?
Turn 6	User Utterance	could you try it for 1 night ? if that works , please book it and get the reference number .
	Generated Belief State	[hotel] {stars = 3, type = hotel, name = gonville hotel, people = 2, day = sunday, stay = 1}; [restaurant] {food = indian, pricerange = expensive}
	Oracle Belief State	[hotel] {stars = 3, type = hotel, people = 2, day = sunday, stay = 1}; [restaurant] {food = indian, pricerange = expensive}
	Delexicalized Response	i was able to successfully book your room . your reference number is [value_reference] . is there anything else i can help you with ?
	Lexicalized Response	i was able to successfully book your room . your reference number is 7rzme01z . is there anything else i can help you with ?
	Oracle Reference	you are all set for 1 night (sunday) , reference number is 6wvv053q . would you like to try another location for monday night ?
Turn 7	User Utterance	no . that is all i need , thank you .
	Generated Belief State	[hotel] {stars = 3, type = hotel, name = gonville hotel, people = 2, day = sunday, stay = 1}; [restaurant] {food = indian, pricerange = expensive}
	Oracle Belief State	[hotel] {stars = 3, type = hotel, people = 2, day = sunday, stay = 1}; [restaurant] {food = indian, pricerange = expensive}
	Delexicalized Response	you are welcome . have a great day !
	Lexicalized Response	you are welcome . have a great day !
	Oracle Reference	thank you , and goodbye .

Table 11: An generated dialogue example from the $\text{PPTOD}_{\text{base}}$ model. (best viewed in color)