Value Imitation Reinforcement Learning in Self-Training Dialogue State Tracking

Jie Yang, Hui Song, Bo Xu and Tianqi Liu

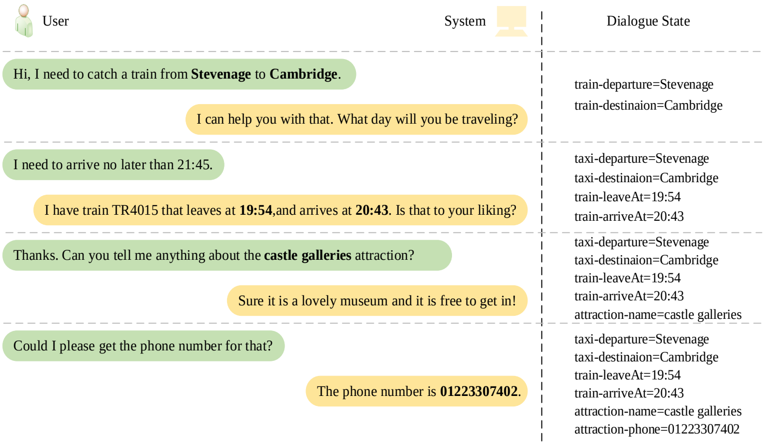
School of Computer Science and Technology, Donghua University, Shanghai, China
{2212525, 2222672}@mail.dhu.edu.cn, {songhui, xubo}@dhu.edu.cn

Abstract. Few-shot Dialogue State Tracking aims to predict the dialogue state with limited labeled data, especially when human annotation is scarce. Existing approaches that use the Self-Training framework often suffer from the gradual drift problem, which results in a noisy expanded labeled dataset. Moreover, except model initialization process, the knowledge of the annotated data has not been fully investigated to accurately deal with unlabeled data. In this paper, we introduce Slot Value Imitation Reinforcement Learning into the Self-Training process to alleviate bias selection and improve the quality of pseudo-label. The reinforcement learning step encourages pseudo-labeled data to imitate the standard value representation of each slot, and then high-confidence pseudo labels are chosen by a dual selection strategy based on value probability and active slot accuracy. Experimental results on the MultiWOZ 2.0 and MultiWOZ 2.4 dataset demonstrate the effectiveness of our proposed model in few-shot DST scenarios. Compared to the original self-training method, Joint Goal Accuracy has a maximum improvement of 2.66% in MultiWOZ 2.0.

Keywords: Dialogue State Tracking, Reinforcement Learning, Self-Training.

1 Introduction

Dialogue State Tracking (DST) tracks the intentions and goals by representing them as a dialogue state, comprising a set of slots and corresponding values, which has shown significant advancements in task-oriented dialogue systems. Fig.1 illustrates some examples of DST within a conversation, where the dialogue state is accumulated and refreshed after each turn. We consider train-departure and train-destination as slots, and their values can be obtained from the current context. In the first turn, their values are Stevenage and Cambridge, respectively. As the turn continues, we can obtain or update the slot values. Traditional supervised methods for training DST models typically rely on large amounts of dialogue corpus and manual annotation. However, data collection and labeling are time-consuming and labor-intensive. Therefore, few-shot DST task has been proposed in order to obtain high-performance DST models with limited annotated corpus and mitigate the issue of data sparsity.



Hi, I need to catch a train from **Stevenage** to **Cambridge**.

I can help you with that. What day will you be traveling?

train-departure=Stevenage
train-destinaion=Cambridge

I need to arrive no later than 21:45.

I have train TR4015 that leaves at **19:54**,and arrives at **20:43**. Is that to your liking?

taxi-departure=Stevenage
taxi-destinaion=Cambridge
train-leaveAt=19:54
train-arriveAt=20:43

Thanks. Can you tell me anything about the **castle galleries** attraction?

Sure it is a lovely museum and it is free to get in!

taxi-departure=Stevenage
taxi-destinaion=Cambridge
train-leaveAt=19:54
train-arriveAt=20:43
attraction-name=castle galleries

Could I please get the phone number for that?

The phone number is **01223307402**.

taxi-departure=Stevenage
taxi-destinaion=Cambridge
train-leaveAt=19:54
train-arriveAt=20:43
attraction-name=castle galleries
attraction-phone=01223307402

Fig. 1. The examples of DST task

Previous works have attempted to leverage large-scale pre-trained language models to tackle the few-shot DST challenge, a task that requires substantial computational resources and access to general text corpora. Although the size of labeled DST data is small, there are abundant and available unlabeled dialogue corpora, which is highly beneficial in practice for the few-shot DST task. [1] proposed to assign pseudo labels for unlabeled data and iteratively enhanced the model's capabilities. In particular, a DST model is first initialized on limited labeled data and used to generate pseudo labels for unlabeled samples, then the pseudo-labeled data was added to labeled dataset and the model will be recurrently trained on the expanded labeled samples. Yet, such Self-Training (ST) strategy is plagued by the problem of gradual drift caused by noisy pseudo labels [2, 3]. The noise in pseudo labels hinders the learning and optimization of the DST model. [4-6] attempted to apply ST on few-shot DST to improve the accuracy. But they either do not handle noisy labels or adopt a simple selection mechanism based solely on predicted probability to obtain high-confidence pseudo labels. Moreover, the scarcity of annotations can lead to selection bias. Consequently, the resulting data may not be clean enough for training, leading to unsatisfactory model accuracy. In DST tasks, active slot accuracy, serving as a key performance metric, measures to the proportion of correctly predicted slots mentioned in a conversation. This metric is able to judge the quality of the predicted dialogue state and objectively measure whether the DST model has successfully finished the user's goals.

To take full advantage of the existing annotation data and exploit the unique characteristics of the DST task, we introduce an explicit slot-aware value feed-back as guidance to strengthen the reliability of pseudo labels. Specifically, in addition to initializing the model with labeled data during self-training procedure, a pseudo-labeled instance is encouraged to mimic the value representation of each slot obtained from the labeled data. This guidance fosters the generation of values in a positive way when predicting pseudo labels. Naturally, Reinforcement Learning (RL) is applicable to encourage this behavior. In our approach, RL is employed to design rewards as feedback signals, steering the DST model to-wards generating more accurate values for the corresponding slot and making precise prediction of the dialogue state. Furthermore, this approach enhances the generalization capability of DST models in few-shot settings.

In this paper, we propose **SUNSET** (involving **S**lot-aware val**U**e imitatio**N** reinforcement learning into Slot-specific s**E**lectable self-**T**raining) to handle few-shot DST. SUNSET is committed to fully exploiting the rich information of unlabeled and labeled data. And the quality of the newly extended labeled dataset is promoted significantly in two phases under a whole self-training framework, that is, during and after the generation step of pseudo labels. To summarize, the main contributions of our work are as follows:

1. We propose a novel self-training process to gradually train a stronger DST model by iteratively allocating highly reliable pseudo labels for unlabeled data. A dual selection strategy based on label probability and active slot accuracy chooses high-confidence pseudo labels into the labeled dataset to mitigate the issue of noise accumulation.
2. We develop a slot-aware value imitation reinforcement learning process that further urges the DST model to produce more exact pseudo labels during self-training procedure by estimating the slot value differences between labeled and unlabeled data.
3. We demonstrate that SUNSET outperforms strong baselines. Extensive experiments and ablation study validate the effectiveness of the proposed method.

2 Related Work

DST constitutes a crucial element within task-oriented dialogue systems. While prior research has demonstrated substantial advancements on benchmark datasets, there remains a dearth of exploration in real-world scenarios. We have limit well-labeled data primarily due to the high cost of annotation. Consequently, zero-shot and few-shot learning have been conducted to tackle such problem and exhibit strong generalization capabilities.

Few-shot DST task gained a lot of attention recently for the reason of the limited annotation data and the diminishing requirement for human supervision. Recent studies have shown that large-scale pretraining language models are effective few-shot learners, as they can leverage external resources to continue to train the language models, thereby adapting knowledge from other Natural Language Processing (NLP) tasks to the DST task [7,8]. For example, TOD-BERT [9] collected nine types of dialogue corpus that contains 100,000 dialogues to further train BERT [10] model. PPTOD [11] collected and organized 11 kinds of dialogue corpora, and then constructed training data from over 2 million discourses and over 80 domains, which will be used to train T5 [12]. DS2 [13] transformed the dialogue state tracking task into a dialogue summary task and collected approximately 200,000 dialogue summary datasets for fine-tuning T5 and BART [14]. However, collecting a large amount of NLP corpora is laborious and expensive computational resources are needed during training. Self-training, a typical semi-supervised learning approach, utilizes limited labeled data to train a base model and allocates pseudo labels to unlabeled instances, steadily expanding the labeled dataset and updating the model. Thus, the model can continue to train on the amplified labeled data to iteratively improve the ability of few-shot DST models. Since the pseudo labels may contain noise, directly adding pseudo labeled data is not a reliable method. GradAug [4] and PPaug [5] calculate the confidence of each pseudo label based on the predicted probability and select those highly confident pseudo data. While CSS [6] did not perform any selection strategy. During each iteration, the model is retrained on a combination of all pseudo-labeled data and ground truth data, and this process is repeated until the model converges. Self-training eliminates the need for manual annotation by leveraging the predictive capability of the model itself, thus mitigating the cost associated with human labor. Yet, the method struggles with the issue of gradual drift, leading to noisy and potentially incorrect pseudo labels.

Reinforcement Learning is a very popular NLP technique recently [15-17]. It facilitates the model to generate correct actions by a reward mechanism during training. However, there are relatively few related researches on DST task. DRQN [18] described a deep reinforcement learning based end-to-end framework for DST and dialog policy tasks. It was evaluated on a 20 Question Game conversational game simulator and produced desirable results. [19] exploited REINFORCE algorithm [20] to optimize end-to-end task-oriented dialogue systems in DSTC2 and movie booking dataset. [21] also applied a deep reinforcement learning framework for on-line DST optimization and achieved promising performance in DSTC2 and DSTC3 datasets.

3 Methodology

As illustrated in Fig.2, the proposed framework **SUNSET** consists of three modules: Base DST Model, **S**lot-aware val**U**e imitatio**N** reinforcement learning step (**SUN**) and **S**lot-specific s**E**lectable self-**T**raining process (**SET**).

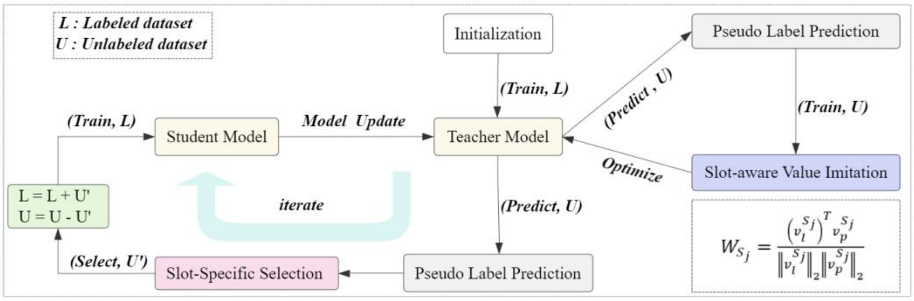


Fig. 2. The overview architecture of SUNSET.

The limited labeled dialogues are first used to derive an initial Teacher Model and then it assigns pseudo labels on abundant unlabeled samples. The SUN step learns a policy that aims to maximize the similarity between the expected value representation of the remaining pseudo-labeled data and the standard representation acquired from labeled data for each slot during the SET process, guiding the Teacher Model towards the correct vector space for value generation. Next, the refined Teacher Model is used in the selection step. Different from the previous selection works, SUNSET not only relies on the predicted label probability, but also focuses on the active slot accuracy for further filtering, as it is a more representative measure of the model's dialog state prediction accuracy. The filtered pseudo-labeled data and initial labeled data are used to train a Student Model from scratch, which then becomes the Teacher Model for the subsequent iteration. Such iteration will continue until all unlabeled dialogues are picked out or the Student Model converges.

3.1 Base DST Model

We employ a baseline DST model based on the BERT value matching framework, shown in Fig. 3. The model comprises two BERT encoders to process dialogue utterances, slots, and values. Additionally, a slot-token attention module is incorporated to capture slot-specific contextual information. Furthermore, a slot-value match module is utilized to calculate scores for candidate values of each slot and the one with highest score will be the final predicted result.

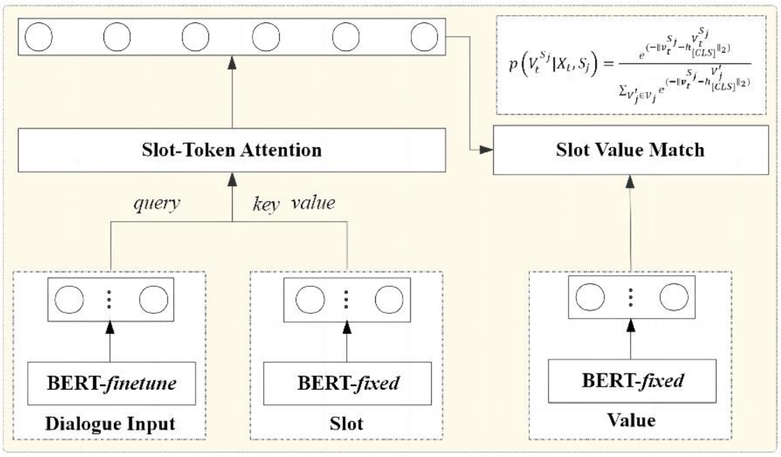

$p\left(V_t^{S_j}|X_t, S_j\right) = \frac{e^{(-\|v_t^{S_j} - h_{[CLS]}^{V_t^{S_j}}\|_2)}}{\Sigma_{V'_j \in \mathcal{V}_j} e^{(-\|v_t^{S_j} - h_{[CLS]}^{V'_j}\|_2)}}$
Slot-Token Attention
Slot Value Match
query
key value
BERT-finetune
Dialogue Input
BERT-fixed
Slot
BERT-fixed
Value

Fig. 3. The framework of Base DST Model.

For turn t, a dialogue utterance can be represented as $Z_t = R_t \oplus U_t$, R_t and U_t represent the system response and user utterance at turn t. And $\oplus$ is the sequence concatenation operator. The dialogue history denotes as $X_t = Z_1 \oplus Z_2 \oplus \cdots \oplus Z_{t-1}$. For j-th slot S_j in the predefined slot set $S = \{S_1, S_2, \cdots, S_J\}$ and its corresponding value $V_t^{S_j}$ in the candidate value space $\mathcal{V}_j$, we can describe a dialog state B_t as several slot-value pairs: $B_t = \{(S_j, V_t^{S_j}) | 1 \leq j \leq J\}$.

We apply a fine-tuned and parameter-fixed BERT model to obtain context and slot and value representations respectively:

$$H_t = BERT_{finetune}([CLS] \oplus U_t \oplus [SEP] \oplus X_t \oplus [SEP]) \tag{1}$$

$$h_{[CLS]}^{S_j} = BERT_{fixed}\big([CLS] \oplus S_j \oplus [SEP]\big) \tag{2}$$

$$h_{[CLS]}^{V_t^{S_j}} = BERT_{fixed}\left([CLS] \oplus V_t^{S_j} \oplus [SEP]\right) \tag{3}$$

Then, a slot-token multi-head attention mechanism captures slot-specific dialogue information. We further feed it into a linear layer and layer normalization to obtain the predicted value representation:

$$c_t^{S_j} = MultiHead\left(h_{[CLS]}^{S_j}, H_t, H_t\right) \tag{4}$$

$$v_t^{S_j} = LN(Linear(c_t^{S_j})) \tag{5}$$

In multi-domain dialogues, the dialogue history typically encompasses intricate and varied information. It is essential to retrieve the most relevant and valuable dialogue information for each slot individually.

Lastly, the DST model predicts the value of each slot S_j via the Euclidean distance between $v_t^{S_j}$ and its candidate value $V_t^{S_j}$. The value with the smallest distance is selected as the final prediction:

$$p\left(V_t^{S_j} | X_t, S_j\right) = \frac{e^{(-\|v_t^{S_j} - h_{[CLS]}^{V_t^{S_j}}\|2)}}{\sum_{V'_j \in \mathcal{V}_j} e^{(-\|v_t^{S_j} - h_{[CLS]}^{V'_j}\|2)}} \tag{6}$$

The distance will be used as one of the confidence scores in the selection strategy. The objective is to maximize the joint probability of all slots with the sum of the negative log-likelihood:

$$\mathcal{L}_t = \sum_{j=1}^{J} -log\left(p\left(V_t^{S_j} | X_t, S_j\right)\right) \tag{7}$$

3.2 Slot-specific Selectable Self-Training

The proposed SUNSET utilizes both labeled data $\mathcal{D}_l = (X_t, B_t)$ with N dialogues as well as unlabeled data $\mathcal{D}_u = (\widetilde{X_t})$ during training. We first initialize a teacher model with $\mathcal{D}_l$. The student model and teacher model all adopt base DST Model framework. Then the teacher model predicts pseudo label $\widetilde{B_t}$ for each unlabeled instance $\widetilde{X_t}$. Before we add the pseudo-labeled dataset $\mathcal{D}_p = (\widetilde{X_t}, \widetilde{B_t})$ to expand original labeled corpus $\mathcal{D}_l$, there is a fresh selection operation on $\mathcal{D}_p$.

Traditionally, the distance $p\left(V_t^{S_j}|X_t,S_j\right)$ calculated by DST model is generally used as the confidence score. For each piece of pseudo data, we calculate the distance score for all slots and take the average as the confidence score for that piece of data:

$$confidence\ score = \frac{1}{J}\sum_{j=1}^{J} p\left(V_t^{S_j}|X_t,S_j\right) \tag{8}$$

The smaller the distance, the higher the confidence level. However, a conversation may not necessarily mention all slots, so we need to concentrate on the slots involved in current dialogue contexts and exploit the accuracy of those active slots as a criterion for predicting a dialogue state. Suppose a dialogue mentions only $J'(1 \le J' \le J)$ slots, the number of slots correctly predicted is $n(1 \le n \le J')$, then the active slot accuracy can be denoted as:

$$active\ accuracy\ = \frac{n}{J'} \tag{9}$$

In this way, our selection strategy can be divided into two stages. Firstly, the top-$k\%$ pseudo-labeled instances from $\mathcal{D}_p$ with highest probability based on Equation (8) will be chosen preliminarily. For hyper-parameter k, we experimentally use 40. Then, we calculate the active slot accuracy of these top-$k\%$ samples separately and the average accuracy. Only those samples that exceeds the average accuracy will be finally picked out, denoting as $\mathcal{D}_s$. After the pseudo labeling and a two-stage selection process, we finally update an extend labeled dataset $\mathcal{D}_l$ with $\mathcal{D}_s$ and use it to train a student model:

$$\mathcal{D}_l \leftarrow \mathcal{D}_l \cup \mathcal{D}_s \tag{10}$$

$$\mathcal{D}_u \leftarrow \mathcal{D}_u - \mathcal{D}_s \tag{11}$$

The best student model for validation dataset will become the teacher model in next iteration. We reinitialize the student model in each iteration to avoid over-fitting. ST procedure will iterate until the student model converges or $\mathcal{D}_u$ becomes depleted.

3.3 Slot-aware Value Imitation Reinforcement Learning

SUNSET aims to minimize the noise and enhance the robustness of the pseudo labels with less labeling biases and errors especially with limited annotations during ST process. To achieve this goal, we individually obtain the value representation under $\mathcal{D}_l$ of each slot when initializing the teacher model as the guideline. The average value representation $v_l^{S_j}$ of slot S_j on $\mathcal{D}_l$ can be computed as:

$$v_l^{S_j} = \frac{\sum v_t^{S_j}}{N_{S_j}} \tag{12}$$

N_{S_j} is the number of conversations in the $\mathcal{D}_l$ that contain slot S_j. Then, assuming that the smaller the distance between the value representation of a pseudo-labeled sample

$(\widetilde{X_t}, \widetilde{B_t})$ and the standard representation, the more accurate the pseudo label will be. In view of the hypothesis, we develop a slot-aware value imitation step with a reinforcement learning framework.

A universal reinforcement learning framework consists of 5 parts: agent, environment, actions, rewards and states. Fig. 4 shows the basic reinforcement learning process. The agent receives the states and rewards from the environment and feeds back a series of actions to the environment.

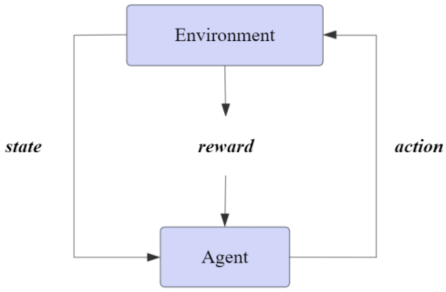


Fig. 4. The reinforcement learning process

Agent: We treat the base DST model as the agent, which will interact with the environment and is responsible for understanding user input and taking actions based on its observations.

Environment: The environment provides feedback to the agent based on its actions. In a dialogue system, changes in the environment are driven by interactions between the user and the system.

Action: The action is to generate a series of slot value pairs given the dialogue context. Such actions have an impact on the environment, leading to a transition from one state to another. The agent's objective is to learn a policy that maps states to actions in a way that maximizes a cumulative reward.

State: In a dialogue system, states refer to the current context of the conversation, including the user's input, the system's response, and the dialogue history. The goal is to empower the agent to make better decisions based on the current state.

Reward: Rewards could be the evaluations of the agent's accuracy that the environment provides to the agent as feedback for its actions. Positive rewards encourage desirable actions, while negative rewards discourage undesirable ones. More concretely, we measure the cosine similarity between $v_l^{s_j}$ and the value representation $v_p^{s_j}$ of a pseudo-labeled data $(\widetilde{X_t}, \widetilde{B_t})$:

$$w_{s_j} = \frac{\left(v_l^{s_j}\right)^T v_p^{s_j}}{\left\|v_l^{s_j}\right\|_2 \left\|v_p^{s_j}\right\|_2} \tag{13}$$

The average cosine distance of all slots in a piece of data is regarded as a reward in he *T*-th time step:

$$W^T = \sum_{j=1}^{J} w_{S_j} \tag{14}$$

For those pseudo-labeled samples with $W^T > \lambda$ (threshold), denoted as $\mathcal{D}_r$, we add them to $\mathcal{D}_l$ and correct the standard value representation corresponding to each slot:

$$\mathcal{D}_l \leftarrow \mathcal{D}_l \cup \mathcal{D}_r \tag{15}$$

$$v_l^{S_j} \leftarrow \frac{1}{N_{S_j}+1}\left(N_{S_j} v_l^{S_j} + v_p^{S_j}\right) \tag{16}$$

In this manner, the positive feedback can attribute the improvement of DST model to produce correct label for next unlabeled data (State). For optimization, the REINFORCE algorithm, a special kind of policy gradient algorithm, is applied to enhance the accuracy of generating pseudo labels. Finally, the DST model will be tuned on each batch using the following loss:

$$\mathcal{L} = \sum_{T=1}^{\mathcal{T}} W^T \times \frac{1}{J} \sum_{j=1}^{J} \left(-log\left(p\left(V_t^{S_j} | X_t, S_j\right)\right)\right) \tag{17}$$

where W^T is the similarity weight, or the reward, log means the negative log-likelihood function. $\mathcal{T}$ denotes the overall time steps in reinforcement learning process and is same as the batch size number. We minimize $\mathcal{L}$ by the gradient descent method to obtain optimal DST model. During the process of calculating loss, SUNSET follows the Markov's decision process and the labeled data $\mathcal{D}_l$ and standard value representation $v_l^{S_j}$ will be dynamically corrected by the selected pseudo-labeled data $\mathcal{D}_r$.

4 Experiment

4.1 Experimental Setup

Dataset. We evaluate our approach on three gradually refined task-oriented dialogue datasets: MultiWOZ 2.0 and the latest MultiWOZ 2.4, containing over 10,000 multi-turn dialogues, covering 7 different domains (*taxi*, *train*, *hotel*, *restaurant*, *attraction*, *hospital* and *police*). MultiWOZ 2.4 mainly corrects the annotation errors in MultiWOZ 2.0. Since hospital and police are not included in the validation set and test set, we use only the remaining 5 domains in the experiments in consistent with previous studies.

Evaluation metric. We compute the Joint Goal Accuracy (JGA) on the test set as the evaluation criteria, which defined over a dataset is the ratio of dialogue turns where all slots have been filled with the correct values according to the ground truth.

Training Details. Following with [17], we randomly select 1%, 5%, 10% and 25% labeled training data to simulate few-shot scenarios. For the ST process, 50% of the training data will be treated as unlabeled data and excluded from the labeled training data. We utilize the pre-trained BERT-base-uncased model as the dialogue context encoder and slot encoder. The training batch size is 4. The maximum input sequence length is set to 512. For slot-token attention module, we set the number of attention heads to 4. We use Adam optimizer and set the warmup proportion to 0.1. The LR decay linearly follows after the warmup phase. The initial learning rate is set to 1e-4 and dropout is 0.3. To determine the optimal threshold λ in the reinforcement learning process, we employed Grid Search across a range of potential values, from 0.3 to 0.7. After extensive evaluation, we identified 0.5 to be the most effective in minimizing the validation error. To optimize computational resource efficiency, each teacher model is trained for 50 epochs and each student model is trained over 6 iterations with 10 epochs for each loop. We run our model several times with different seeds in our machine and finally the average performance is presented.

4.2 Comparison Models

We compare our model with the following few-shot DST models.

- **TRADE** [22] is an initial model that experimented on the MultiWOZ dataset in zero-shot and few-shot settings.
- **TRADE+SS** preserves latent consistency by utilizing stochastic word dropout to be robust to unseen data.
- **MinTL** [23] leverages a plug-and-play pre-trained seq2seq model to jointly learn DST and response generation.
- **STAR** [24] propose a slot self-attention mechanism that can learn the slot correlations automatically.
- **TOD-BERT** is a BERT-based model trained on 9 public human-human dialogue datasets.
- **DS2** reformulates DST as a dialogue summarization task using state-to-summary templates and summary-to-state converter.
- **CSS** combines self-training and contrastive self-supervised to train a DST model. It expands all predicted pseudo-labels to the initial labeled dataset and is then used to train the student model.
- **GradAug** selects the top-*k* instances from pseudo-labels with the highest cosine similarity and a text augmentation technique to expand labeled dataset for predefined ontology DST.
- **PPaug** chooses top-*k* instances with the average softmax value for generative DST models and a data augmentation method is proposed to get more accurate pseudo label.
- **ST-DST** is the Base DST Model introduced in section 3.1 with an original self-training process without any selection and reinforcement learning algorithm. Fig.5 shows its architecture. The predicted pseudo-labeled data in the ST-DST is directly added to the annotated dataset to train the student model. In the next iteration, it will continue to predict the same unlabeled dataset. Until the student model converges, the iteration terminates.

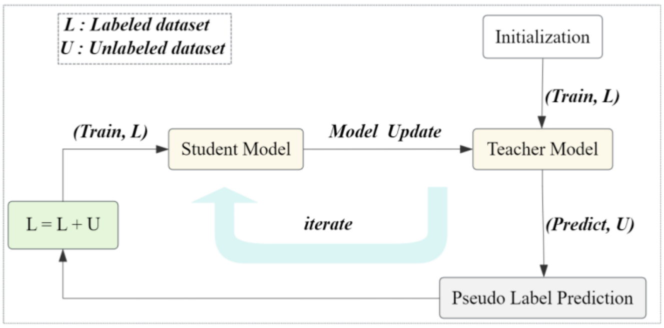


Fig. 5. The overview of ST-DST

4.3 Main Results

Table 1 shows the JGA of our model in comparison to various few-shot DST models on the test set of MultiWOZ 2.0. Because there is limited research on conducting four different scenario few-shot experiments on the MultiWOZ 2.4 dataset, we did not delve into this dataset extensively. Our exploration on this dataset was confined to the ablation experiments, where we investigated the performance of our model.

Table 1. The JGA on MultiWOZ 2.0 in 4 few-shot cases.

Model	Pre-trained Model	JGA (%)			
		1%	5%	10%	25%
TRADE	-	9.70	29.38	34.07	41.41
MinTL	BART-large	9.25	21.28	30.32	-
STAR	BERT-base	8.08	26.41	38.45	48.29
TOD-BERT	BERT-base	10.30	27.80	38.80	44.30
CSS	BERT-base	14.06	41.14	47.96	51.88
GradAug	TOD-BERT	9.9	-	28.3	-
PPaug	T5-small	-	-	44.09	-
DS2	T5-large	**36.15**	**45.14**	47.61	-
ST-DST	BERT-base	15.47	41.41	47.83	52.28
SUNSET	BERT-base	16.45	43.31	**50.49**	**53.69**

As can be seen, our approach outperforms all the baseline models with fewer parameters and achieves the best performance in 10-shot and 25-shot settings. In the 1-shot and 5-shot cases, SUNSET exhibits a slight inferiority to DS2 and the reason maybe that it uses a pre-trained language model with larger parameters and stronger learning capability and utilizes external data from the dialogue summary task. Specially, compared with ST-DST, SUNSET performs much better and the JGA improved by 0.98%, 1.9%, 2.66%, and 1.41% in four different shot cases respectively, which indicates the remarkable contribution of reinforcement learning framework and self-training selection strategy. The clean pseudo-labeled data, contributes to effective model learning, completed by reinforcement strategies employed during the learning process, significantly enhancing training efficiency. It is worth noting that TOD-BERT [9] introduces external dialogue annotation data to train a language model, yet it still does not perform as well as our model and the performance has lagged significantly behind. Both PPaug and GradAug adopt a data augmentation technique by leveraging masked language models to generate more accurate pseudo labels during self- training process and further enhancing the accuracy of self- training process. Even though they use larger pre-trained models, they are less effective than SUNSET in some few-shot cases. Besides, compared to MinTL, STAR and TOD-BERT, SUNSET outperforms those approaches utilizing pre-trained models with similar parameter scales.

4.4 Ablation Study

In order to further validate the promotion of SUN and SET on few-shot DST, we conduct an ablation experiment on MultiWOZ 2.0 and MultiWOZ 2.4, and explore the effects of the two modules separately using 25% training data. We can observe from Table 2, ST-DST w/ SUN and ST-DST w/ SET both increase the baseline JGA to vary degrees and show more significant improvement when used together (SUNSET) in two datasets. The train data size represents the size of the new training dataset, which is composed of the currently selected pseudo-labeled data and the initial label data.

Table 2. Ablation study on MultiWOZ 2.0 and MultiWOZ 2.4 in 25-shot DST.

Model	MultiWOZ 2.0		MultiWOZ 2.4	
	Train data size	JGA (%)	Train data size	JGA (%)
ST-DST	41359	52.28	41361	71.09
ST-DST w/ SUN	41359	52.42	41361	71.59
ST-DST w/ SET	29518	53.00	27212	71.21
SUNSET	32100	53.69	27719	71.96

Concretely, ST-DST w/ SUN does not have a selection strategy in contrast to SUNSET, yet the performance gains show that the reinforcement learning method is able to boost Teacher Model to predict more accurate pseudo labels, which will be used in subsequent self-training process. Fig. 6 draws the tendency of the JGA of pseudo-labeled data during the self-training iteration on MultiWOZ 2.0 and MultiWOZ 2.4 dataset in 25-shot setting. As the iteration goes on, the JGA on unlabeled data increases. The main purpose of SUN is to guide Teacher Model to generate less noisy pseudo labels (a set of slot value pair) using labeled dataset and optimize label consistency. SUN minimizes the spatial distance between the standard value vector obtained from labeled data and the value representation of the corresponding slot for each pseudo data. The improvement in effectiveness indicates that the model has learned to imitate the value representation on labeled data when generating pseudo label, and strives to generate slot values along this direction.

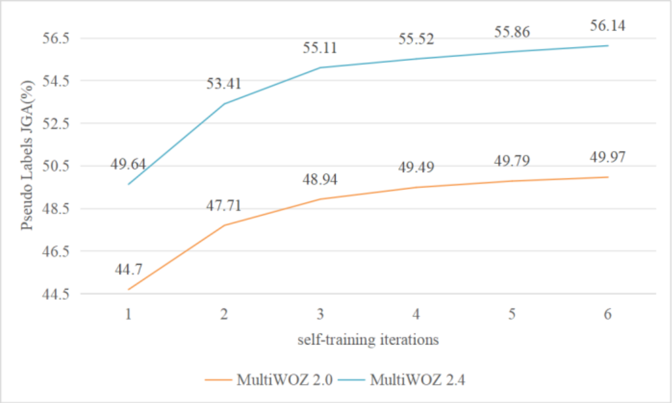


Fig. 6. The tendency of JGA of pseudo-labeled data on two datasets.

ST-DST w/ SET chooses the pseudo labels with high confidence, thus further reducing the noise of the augmented labeled dataset. Besides, it utilizes less training data while achieving higher JGA. We should clearly state that ST-DST model does not exploit any selection strategy on pseudo-labeled data, leading to too much noise in the dataset and interfering with the learning process, which may be one of the reasons for its low JGA. The same conclusion can be drawn from the comparison between the CSS model and the SUNSET model in Table 1.

SUN enhances the quality of the generated pseudo labels by strengthening their reliability during generating, while SET further picks out the effective pseudo labels, and only the reliable pseudo-labels are finally augmented. Thus, SUNSET gets the newly training dataset from two different perspectives and diminishes error accumulation effectively, facilitating the learning of Student Model.

5 Conclusion

In this paper, we propose a novel semi-supervised model SUNSET involving reinforcement learning into self-training process for low resource DST. SUNSET seeks to boost the reliability of pseudo labels and obtains a more refined labeled dataset from two different perspectives. On the one hand, the two-stage selection strategy for pseudo labels based on the probability of predicted values and the active slot accuracy will filter out noisy samples. On the other hand, SUNSET encourages pseudo-labeled data to imitate the standard value representation of labeled data under a policy gradient optimization algorithm, which enables the DST model to yield more precise pseudo dialogue states. Our proposed model effectively alleviates the gradual drift problem suffered by traditional self-training methods. Experiments on two dialogue benchmarks demonstrate the effectiveness of SUNSET in the few-shot DST scenario, showcasing its ability to improve performance with limited labeled data.

References

1. Rosenberg, C., Hebert, M., Schneiderman, H.: Semi-supervised self- training of object detection models. In: IEEE Workshop on Applications of Computer Vision (2005)
2. Curran, J., Murphy, T., Scholz, B.: Minimising semantic drift with mutual exclusion bootstrapping (2007)
3. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires rethinking generalization (2016)
4. Mi, F., Zhou, W., Cai, F., Kong, L., Huang, M., Faltings, B.: Self- training improves pretraining for few-shot learning in task-oriented dialog systems. arXiv preprint arXiv:2108.12589 (2021)
5. Lee, J., Lee, C., Kim, Y., Lee, G.G.: Self-training with purpose preserving augmentation improves few-shot generative dialogue state tracking. arXiv preprint arXiv:2211.09379 (2022)
6. Zhang, H., Bao, J., Sun, H., Luo, H., Li, W., Cui, S.: Css: Combining self-training and self-supervised learning for few-shot dialogue state tracking. arXiv preprint arXiv:2210.05146 (2022)
7. Gao, S., Agarwal, S., Chung, T., Jin, D., Hakkani-Tur, D.: From machine reading comprehension to dialogue state tracking: Bridging the gap. arXiv preprint arXiv:2004.05827 (2020)
8. Lin, Z., Liu, B., Madotto, A., Moon, S., Crook, P., Zhou, Z., Wang, Z., Yu, Z., Cho, E., Subba, R., et al.: Zero-shot dialogue state tracking via cross-task transfer. arXiv preprint arXiv:2109.04655 (2021)
9. Wu, C.S., Hoi, S., Socher, R., Xiong, C.: Tod-bert: Pre-trained natural language understanding for task-oriented dialogue. arXiv preprint arXiv:2004.06871 (2020)
10. Jacob D., et al.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019: 4171-4186.
11. Su, Y., Shu, L., Mansimov, E., Gupta, A., Cai, D., Lai, Y.A., Zhang, Y.: Multi-task pretraining for plug-and-play task-oriented dialogue system. arXiv preprint arXiv:2109.14739 (2021)
12. Raffel C, Shazeer N, Roberts A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. The Journal of Machine Learning Research, 2020, 21(1): 5485-5551.
13. Shin, J., Yu, H., Moon, H., Madotto, A., Park, J.: Dialogue summaries as dialogue states (ds2), template-guided summarization for few-shot dialogue state tracking. arXiv preprint arXiv:2203.01552 (2022)
14. Lewis M, Liu Y, Goyal N, et al. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension[C]. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 7871-7880.
15. Li, J., Monroe, W., Ritter, A., Galley, M., Gao, J., Jurafsky, D.: Deep reinforcement learning for dialogue generation. arXiv preprint arXiv:1606.01541 (2016)
16. Su, P.H., Gasic, M., Mrksic, N., Rojas-Barahona, L., Ultes, S., Vandyke, D., Wen, T.H., Young, S.: On-line active reward learning for policy optimisation in spoken dialogue systems. arXiv preprint arXiv:1605.07669 (2016)
17. Yu, L., Zhang, W., Wang, J., Yu, Y.: Seqgan: Sequence generative adversarial nets with policy gradient. In: Proceedings of the AAAI conference on artificial intelligence. vol. 31 (2017)

18. Zhao, T., Eskenazi, M.: Towards end-to-end learning for dialog state tracking and management using deep reinforcement learning. arXiv preprint arXiv:1606.02560 (2016)
19. Liu, B., Tur, G., Hakkani-Tur, D., Shah, P., Heck, L.: End-to-end optimization of task-oriented dialogue model with deep reinforcement learning. arXiv preprint arXiv:1711.10712 (2017)
20. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Reinforcement learning pp. 5-32 (1992)
21. Chen, Z., Chen, L., Zhou, X., Yu, K.: Deep reinforcement learning for on-line dialogue state tracking. In: National Conference on Man-Machine Speech Communication. pp. 278-292. Springer (2023)
22. Wu, C.S., Madotto, A., Hosseini-Asl, E., Xiong, C., Socher, R., Fung, P.: Transferable multi-domain state generator for task-oriented dialogue systems. arXiv preprint arXiv:1905.08743 (2019)
23. Lin, Z., Madotto, A., Winata, G.I., Fung, P.: Mintl: Minimalist transfer learning for task-oriented dialogue systems. arXiv preprint arXiv:2009.12005 (2020)
24. Ye, F., Manotumruksa, J., Zhang, Q., Li, S., Yilmaz, E.: Slot self-attentive dialogue state tracking. In: Proceedings of the Web Conference2021. pp. 1598-1608 (2021)