CSS: Combining Self-training and Self-supervised Learning for Few-shot Dialogue State Tracking

Haoning Zhang[1,3], **Junwei Bao**[2*], **Haipeng Sun**[2],
Huaishao Luo[2], **Wenye Li**[4], **Shuguang Cui**[3,1,5]
[1]FNii, CUHK-Shenzhen [2]JD AI Research
[3]SSE, CUHK-Shenzhen [4]SDS, CUHK-Shenzhen [5]Pengcheng Lab
haoningzhang@link.cuhk.edu.cn, sunhaipeng6@jd.com,
{baojunwei001, huaishaoluo}@gmail.com,
{wyli, shuguangcui}@cuhk.edu.cn

Abstract

Few-shot dialogue state tracking (DST) is a realistic problem that trains the DST model with limited labeled data. Existing few-shot methods mainly transfer knowledge learned from external labeled dialogue data (e.g., from question answering, dialogue summarization, machine reading comprehension tasks, etc.) into DST, whereas collecting a large amount of external labeled data is laborious, and the external data may not effectively contribute to the DST-specific task. In this paper, we propose a few-shot DST framework called **CSS**, which **C**ombines **S**elf-training and **S**elf-supervised learning methods. The unlabeled data of the DST task is incorporated into the self-training iterations, where the pseudo labels are predicted by a DST model trained on limited labeled data in advance. Besides, a contrastive self-supervised method is used to learn better representations, where the data is augmented by the dropout operation to train the model. Experimental results on the MultiWOZ dataset show that our proposed CSS achieves competitive performance in several few-shot scenarios.[1]

1 Introduction

Dialogue state tracking (DST) is an essential sub-task in a task-oriented dialogue system (Yang et al., 2021; Ramachandran et al., 2022; Sun et al., 2022). It predicts the dialogue state corresponding to the user's intents at each dialogue turn, which will be used to extract the preference and generate the natural language response (Williams and Young, 2007; Young et al., 2010; Lee and Kim, 2016; Mrkšić et al., 2017; Xu and Hu, 2018; Wu et al., 2019a; Kim et al., 2020; Ye et al., 2021; Wang et al., 2022). Figure 1 gives an example of DST in a conversation, where the dialogue state is accumulated and updated after each turn.

---

*Corresponding author
[1]Our code is available at https://github.com/JD-AI-Research-NLP/CSS

Usr:	Hi I am looking for a **restaurant in the north** that serves **Asian** oriental food.
Sys:	I would recommend **Saigon city**. Would you like to make a reservation?
Usr:	That sounds great! We would like a reservation for **Monday** at **16:45** for **6 people**. Can I get the reference number for our reservation?

restaurant-area-north	restaurant-food-Asian
restaurant-name-Saigon city	restaurant-book day-Monday
restaurant-book time-16:45	restaurant-book people-6

Table 1: A dialogue example containing utterances from user and system sides and the corresponding dialogue state (a set of domain-slot-value pairs).

Training a DST model requires plenty of dialogue corpus containing dialogue utterances and human-annotated state labels, whereas annotating is costly. Therefore, the DST models are expected to have acceptable performance when trained with limited labeled data, i.e., in the few-shot cases (Wu et al., 2020b). Previous studies on few-shot DST solve the data scarcity issue mainly by leveraging external labeled dialogue corpus to pre-train the language models, which are then transferred into the DST task (Wu et al., 2020a; Su et al., 2022; Shin et al., 2022). However, there exist several disadvantages: first, collecting a large amount of external labeled data is still laborious; second, utilizing the external data is heavily dependent on computational resources since the language models have to be further pre-trained; third, the external data always comes from different conversation scenarios and NLP tasks, such as dialogues in multi topics, question answering, dialogue summary, etc. The data types and distributions differ from the DST-specific training data, making it less efficient to transfer the learned knowledge into DST.

We consider utilizing the unlabeled data of the DST task, which is easy to access and has similar contents to the limited labeled data, so that the DST model can be enhanced by training on an enlarged amount of data corpus. In this pa-
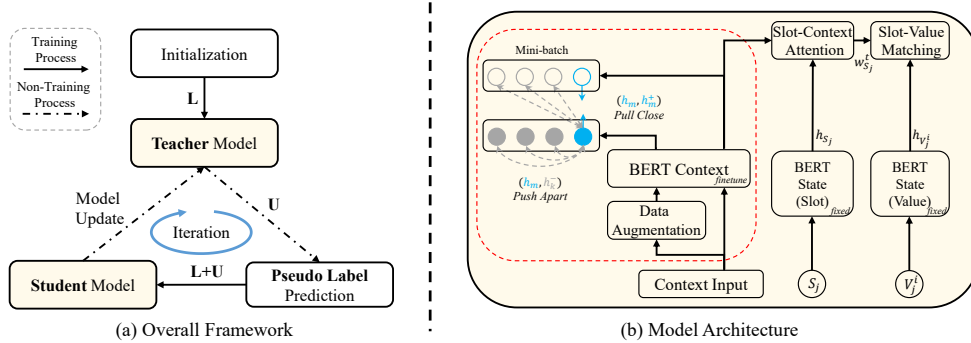
Figure 1: The description of CSS. Part (a) is the overall teacher-student training iteration process, **L** and **U** correspond to labeled and unlabeled data. Part (b) is the model architecture for both teacher and student, where the red dashed box is the illustration of the self-supervised learning object through the dropout augmentation: narrow the distance between each instance and its corresponding augmented one (pull close), enlarge its distance to the rest in the same batch in representation area (push apart).

per, we propose a few-shot DST framework called CSS, which **C**ombines the **S**elf-training and **S**elf-supervised methods. Specifically, a DST model is first trained on limited labeled data and used to generate the pseudo labels of the unlabeled data; then both the labeled and unlabeled data can be used to train the model iteratively. Besides, we augment the data through the contrastive self-supervised dropout operation to learn better representations. Each training instance is masked through a dropout embedding layer, which will act as the contrastive pair, and the model is trained to pull the original and dropout instances closer in the representation area. Experiments on the multi-domain dialogue dataset MultiWOZ demonstrate that our CSS achieves competitive performance with existing few-shot DST models.

2 Related Work

Few-shot DST focuses on the model performance with limited labeled training data, which overcomes the general data scarcity issue. Existing DST models enhance the few-shot performance mainly by incorporating external data of different tasks to further pre-train a language model, which is still collection and computational resources demanding (Gao et al., 2020; Lin et al., 2021; Su et al., 2022; Shin et al., 2022). Inspired by self-training that incorporates predicted pseudo labels of the unlabeled data to enlarge the training corpus (Wang et al., 2020; Mi et al., 2021; Sun et al., 2021), in this paper, we build our framework upon the NoisyStudent method (Xie et al., 2020) to enhance the DST model in few-shot cases.

Self-supervised learning trains a model on an auxiliary task with the automatically obtained ground-truth (Mikolov et al., 2013; Jin et al., 2018; Wu et al., 2019b; Devlin et al., 2019; Lewis et al., 2020). As one of the self-supervised approaches, contrastive learning succeeds in various NLP-related tasks, which helps the model learn high-quality representations (Cai et al., 2020; Klein and Nabi, 2020; Gao et al., 2021; Yan et al., 2021). In this paper, we construct contrastive data pairs by the dropout operation to train the DST model, which does not need extra supervision.

3 Methology

Figure 1 shows the CSS framework, where (a) is the overall training framework, and (b) is the architecture of both teacher and student models. Our CSS follows the NoisyStudent self-training framework (Xie et al., 2020). After deriving a teacher DST model trained with labeled data, it's continuously trained and updated into the student DST model with both labeled and unlabeled data, where the pseudo labels of the unlabeled data are synchronously predicted. Unlike the original NoisyStudent augmenting training data only in the student training stage, we implement the contrastive self-supervised learning method in both training teacher and student models, where each training instance is augmented through a dropout operation, and the model is trained to group each instance with its augmented pair closer and diverse it far from the rest in the same batch.

3.1 DST Task and Base Model

Let's define $D_t = \{(Q_t, R_t)\}_{t=1:T}$ as the set of system query and user response pairs in total T

turns, B_t as the dialogue state for each dialogue turn, which contains a set of (domain-slot S, value V) pairs: $B_t = \{(S_j, V_j^i)|1 \le j \le J, 1 \le i \le I\}$, assuming there are J (domain-slot) pairs, and $\mathcal{V}_j = \{V_j^i\}$ is the value space of slot S_j with I candidates. DST task aims to generate the dialogue state at the t-th turn B_t, given all the dialogue utterances and the predicted state from the previous turn.

The base DST model is a standard BERT-based matching framework training on a small dataset (Ye et al., 2022), denoted as **BASE**. The context input is the concatenation of the dialogue utterances and state from the previous turn, denoted as $C_t = [CLS] \oplus D_1 \oplus ... \oplus D_{t-1} \oplus B_{t-1} \oplus [SEP] \oplus D_t \oplus [SEP]$; a BERT context encoder encodes the context input, denoted as $H_t = BERT_{finetune}(C_t)$; for slots and values, another BERT state encoder with fixed parameters is used to derive the representations: $h_{S_j} = BERT_{fixed}(S_j)$, $h_{V_j^i} = BERT_{fixed}(V_j^i)$. During training, the parameters of the BERT state encoder will not be fine-tuned. For each slot, its context-relevant feature is derived through the multi-head attention, where the slot representation acts as query, the context representation acts as both the key and value (Vaswani et al., 2017): $r_{S_j}^t = MultiHead(h_{S_j}, H_t, H_t)$. Then it's transformed by a linear and normalization layer: $w_{S_j}^t = LayerNorm(Linear(r_{S_j}^t))$, which is used to calculate the distance with each value representation of S_j, and the one with the smallest distance will be selected. The probability of selecting the ground truth $h_{V_j^{i'}}$ is denoted as:

$$P(V_j^{i'}|C_t, S_j) = \frac{exp(-||w_{S_j}^t - h_{V_j^{i'}}||_2)}{\sum_{V_j^i \in \mathcal{V}_j} exp(-||w_{S_j}^t - h_{V_j^i}||_2)}, \tag{1}$$

and the DST objective is to minimize the sum of the negative log-likelihood among the J slots:

$$L_d = \sum_{j=1}^{J} -log(P(V_j^{i'}|C_t, S_j)). \tag{2}$$

We implement our CSS built on the model BASE, and it's also available to transfer CSS into other DST-related models.

3.2 Self-training

Let L, U be labeled and unlabeled data, $\mathbf{X} = \{x_n\}_{n=1:N}$ be the set of training instances containing N dialogues. A teacher f_T is trained with L; then for each $x_n \in U$, the dialogue state is predicted by f_T and acts as the pseudo label. Both

L (with ground labels) and U (with pseudo labels) are used to train a student f_S with the following objective function:

$$L_d^* = \sum_{j=1}^{J} -log(P(V_j^{i*}|C_t, S_j))$$
$$+ \sum_{j=1}^{J} -log(P(V_j^{i'}|C_t, S_j)) \tag{3}$$

V_j^{i*} and $V_j^{i'}$ correspond to the pseudo and ground labels from U and L. f_S will replace f_T to re-predict the pseudo labels on U, and the training-prediction-training loop will iterate until f_S converges.

3.3 Self-supervised Learning

We implement the contrastive self-supervised method to learn better representations, where a simple yet effective dropout operation augments the training instances. Denote $\{x_m\}_{m=1:M}$ as the training instances in a batch with size M. Each x_m is augmented into x_m^+ through a dropout embedding layer, and both of them are encoded by the BERT context encoder: $h_m = BERT_{finetune}(x_m)$, $h_m^+ = BERT_{finetune}(x_m^+)$. Then the model is trained to narrow their representation distances with the contrastive objective:

$$L_m = -log\frac{e^{sim(h_m, h_m^+)/\tau}}{e^{sim(h_m, h_m^+)/\tau} + \sum_{k=1}^{2M-2}(e^{sim(h_m, h_k^-)/\tau})}, \tag{4}$$

where τ is the temperature parameter, and $\{h_k^-\}$ correspond to all training instances in the same batch except h_m and h_m^+ ($2M - 2$ instances). In simpler words, each training instance and its dropout pair are treated as the ones having similar semantic representations.

3.4 Optimization

Besides the CSS and BASE models, another two ablations on BASE are conducted: BASE w/ SSL and BASE w/ ST. We first train a BASE model: $L_{BASE} = L_d$, then we train a BASE model adding the self-supervised method, denoted as BASE w/ SSL: $L_{SSL} = L_d + L_m$. Next the unlabeled data is incorporated, and we train a BASE model adding the self-training iterations, denoted as BASE w/ ST: $L_{ST} = L_d^*$, and finally we train the CSS model: $L_{CSS} = L_d^* + L_m$. The performance of the four models will be shown in Section 4.

Models	Pre-trained Model (# Params.)	1%	5%	10%	25%	100%
TRADE (Wu et al., 2019a)	-	9.70	29.38	34.07	41.41	48.62
MinTL (Lin et al., 2020)	BART-large (400M)	9.25	21.28	30.32	-	52.10
TRADE$_{ssup}$ (Wu et al., 2020b)	-	20.41	33.67	37.16	42.69	48.72
STAR (Ye et al., 2021)	BERT-base (110M)	8.08	26.41	38.45	48.29	54.53
TOD-BERT* (Wu et al., 2020a)	BERT-base (110M)	10.30	27.80	38.80	44.30	-
PPTOD* (Su et al., 2022)	T5-large (770M)	31.46	43.61	45.96	49.27	53.89
DS2* (Shin et al., 2022)	T5-large (770M)	**36.15**	**45.14**	47.61	50.45	54.78
BASE	BERT-base (110M)	13.19	37.19	44.23	49.20	53.97
CSS	BERT-base (110M)	14.06	41.90	**47.96**	**51.88**	**55.02**

Table 2: Joint goal accuracy on MultiWOZ 2.0. * means the model incorporates external labeled dialogue data to pre-train a language model. The results of TRADE and TOD-BERT come from Wu et al. (2020b); MinTL comes from Su et al. (2022); STAR, 25% of PPTOD and DS2 are reproduced by using their released codes.

4 Experiments

In this section, we first give the experimental dataset and training details, then show the experimental results compared with several existing baselines, ablation studies in both multi-domain and single-domain accuracy, and the error analysis.

4.1 Dataset and Few-shot Settings

We evaluate our CSS on MultiWOZ 2.0 (Budzianowski et al., 2018), a task-oriented dialogue dataset containing 7 domains (attraction, hospital, hotel, police, restaurant, taxi, train) and around 8400 multi-turn training dialogues. Since the hospital and police domains do not have dialogues in validation and test sets, we follow the previous work (Wu et al., 2019a) to use five domains (attraction, hotel, restaurant, taxi, train) as training data with 30 (domain, slot) pairs. We randomly select 1%, 5%, 10%, and 25% labeled training data to simulate the few-shot cases. For self-training, the amount of unlabeled data is 50% of the training dataset in MultiWOZ 2.0 and excluded from the labeled training data. For each case, we use three different fixed random seeds during the whole data selection and training process, and the final result is averaged. We use the joint goal accuracy to evaluate the model, which is the ratio of dialogue turns that all the (domain-slot-value) pairs are correctly predicted.

4.2 Training Details

We choose BERT-base-uncased as the context encoder. The batch size is set to 8. The AdamW optimizer is applied to optimize the model with the learning rate 4e-5 and 1e-4 for encoder and decoder (Loshchilov and Hutter, 2019). Both the dropout rate and the temperature parameter are set to 0.1.

All the models are trained on a single P40. For the sake of the computation resources efficiency, each teacher DST model is trained for 50 epochs, and each student model is trained over 3 iteration loops with 10 epochs for each loop.

4.3 Main Results

Table 2 shows the results in terms of joint goal accuracy. Our CSS generally performs well in the four few-shot settings, especially achieving SOTA results using 10% and 25% training data. Besides, CSS outperforms all the methods using 100% labeled training data, where all the labeled dialogues are used to train a teacher model, and the student model is trained on 150% data, 50% of which has both labels and pseudo labels. It's also observed that when using 1% and 5% training data, PPTOD and DS2 perform better than others. Specifically, both PPTOD and DS2 use the T5-large language model (Raffel et al., 2020), which has a remarkable contribution to the prediction accuracy, especially when the amount of labeled DST data is strictly limited. Besides, PPTOD pre-trains T5 on various dialogue-related tasks and data, and DS2 also pre-trains T5 on dialogue summarization data, which further enhance their DST models by the dialogue-related knowledge. Therefore, compared with them, we conclude that the superiority of our CSS mainly comes from efficiently utilizing the DST-related unlabeled data, instead of the large language model or external dialogue data.

4.4 Ablation Studies

Table 3 shows the performance of four models: BASE, BASE w/ SSL, BASE w/ ST, and CSS, in terms of joint goal accuracy. Table 4 shows the joint accuracy for every single domain using 5% training data. It can be observed that in both two tables,

	1%	5%	10%	25%
BASE	13.19	37.19	44.23	49.20
w/ SSL	13.26	38.73	44.87	49.48
w/ ST	14.07	40.33	47.00	51.78
CSS	14.06	41.90	47.96	51.88

Table 3: Joint goal accuracy on MultiWOZ 2.0 of CSS and three ablations: BASE, BASE w/ SSL (self-supervised learning), BASE w/ ST (self-training).

5%	attraction	hotel	restaurant	taxi	train
BASE	60.26	48.80	53.17	63.12	72.60
w/ SSL	60.98	50.74	53.22	62.88	75.24
w/ ST	61.34	51.96	55.69	63.12	77.41
CSS	62.97	53.27	57.32	63.55	79.13

Table 4: Domain joint accuracy using 5% labeled training data.

BASE w/ SSL and BASE w/ ST perform better than BASE, and CSS gets the best accuracy, indicating the effectiveness of each individual method. Detailed experiment results are shown in A.1.

4.5 Error Analysis

We further analyze the wrong prediction types. There are 3 wrong types. Type I means the model fails to predict a correct (domain-slot-value) pair (the predicted value is *none* while the ground truth is not, denoted as *active*), Type II means the model predicts a value not contained in the ground truth (the ground truth value is *none*), and Type III means the model predicts a value different from the ground truth (both the predicted and ground truth value are *active*). We use the CSS model trained on 5% labeled training data to make predictions on the testset. Among the dialogue turns containing wrong predictions, we randomly sample 100 turns and then sum all the wrong predicted domain-slot-value pairs, which is 203 in total. Table 5 shows the comparison of three wrong types, the number of each wrong type pairs and the corresponding ratio, where Type I is the most common case. Table 6 gives a three-turn dialogue example containing wrong predictions. This indicates that the prediction performance can be further enhanced by better modeling the dialogue context from history turns, and we leave it in further studies.

5 Conclusion

In this paper, we propose CSS, a training framework combining self-training and self-supervised

Error Type	Ground	Prediction	Count	Ratio
I	*active*	*none*	88	43.35%
II	*none*	*active*	68	33.50%
III	*active*	*active*	47	23.15%

Table 5: The comparison of three wrong types and the number and corresponding ratio of wrong predictions in 100 sampled turns (203 wrong predictions in total).

Usr (turn 1)	I need a taxi at **Lan Hong House** to leave by **14:45**.
Sys (turn 2)	Okay, what is your destination?
Usr (turn 2)	I want to go to the **Leicester** train station.
Sys (turn 3)	Have you in a white Honda, 07040297067 is the phone number.
Usr (turn 3)	Thanks for the quick response.
Ground	taxi-departure-Lan Hong House taxi-leaveat-14:45 taxi-destination-Leicester
Prediction	train-destination-Leicester, taxi-destination-Autumn House

Table 6: A dialogue example containing three turns (divided by short lines) and the wrong predicted dialogue state at the third turn (the slots with value *none* are omitted). The blue, orange, red domain-slot-value pairs correspond to the wrong type I, II, III.

learning for the few-shot DST task. The self-training enlarges the training data corpus by incorporating unlabeled data with pseudo labels to train a better DST model, and the contrastive self-supervised learning method helps learn better representations without extra supervision. Compared with the previous methods leveraging knowledge learned from a large amount of external labeled dialogue data, CSS is superior in smaller data scales and less computational resources. Experiments on MultiWOZ 2.0 demonstrate the effectiveness of CSS in several few-shot scenarios.

Acknowledgements

References

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Hengyi Cai, Hongshen Chen, Yonghao Song, Zhuoye Ding, Yongjun Bao, Weipeng Yan, and Xiaofang Zhao. 2020. Group-wise contrastive learning for neural dialogue generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 793–802, Online. Association for Computational Linguistics.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shuyang Gao, Sanchit Agarwal, Di Jin, Tagyoung Chung, and Dilek Hakkani-Tur. 2020. From machine reading comprehension to dialogue state tracking: Bridging the gap. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 79–89, Online. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xisen Jin, Wenqiang Lei, Zhaochun Ren, Hongshen Chen, Shangsong Liang, Yihong Zhao, and Dawei Yin. 2018. Explicit state tracking with semi-supervisionfor neural dialogue generation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 1403–1412. ACM.

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*, pages 567–582, Online. Association for Computational Linguistics.

Tassilo Klein and Moin Nabi. 2020. Contrastive self-supervised learning for commonsense reasoning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7517–7523, Online. Association for Computational Linguistics.

Byung-Jun Lee and Kee-Eung Kim. 2016. Dialog history construction with long-short term memory for robust generative dialog state tracking. *Dialogue & Discourse*, 7(3):47–64.

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*, pages 7871–7880, Online. Association for Computational Linguistics.

Zhaojiang Lin, Bing Liu, Andrea Madotto, Seungwhan Moon, Zhenpeng Zhou, Paul Crook, Zhiguang Wang, Zhou Yu, Eunjoon Cho, Rajen Subba, and Pascale Fung. 2021. Zero-shot dialogue state tracking via cross-task transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7890–7900, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Fei Mi, Wanhao Zhou, Lingjing Kong, Fengyu Cai, Minlie Huang, and Boi Faltings. 2021. Self-training improves pre-training for few-shot learning in task-oriented dialog systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1887–1898, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *ArXiv preprint*, abs/1301.3781.

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.

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

Govardana Sachithanandam Ramachandran, Kazuma Hashimoto, and Caiming Xiong. 2022. [CASPI] causal-aware safe policy improvement for task-oriented dialogue. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 92–102, Dublin, Ireland. Association for Computational Linguistics.

Jamin Shin, Hangyeol Yu, Hyeongdon Moon, Andrea Madotto, and Juneyoung Park. 2022. Dialogue summaries as dialogue states (DS2), template-guided summarization for few-shot dialogue state tracking. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3824–3846, Dublin, Ireland. Association for Computational Linguistics.

Yixuan Su, Lei Shu, Elman Mansimov, Arshit Gupta, Deng Cai, Yi-An Lai, and Yi Zhang. 2022. Multi-task pre-training for plug-and-play task-oriented dialogue system. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4661–4676, Dublin, Ireland. Association for Computational Linguistics.

Haipeng Sun, Junwei Bao, Youzheng Wu, and Xiaodong He. 2022. BORT: Back and denoising reconstruction for end-to-end task-oriented dialog. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2156–2170, Seattle, United States. Association for Computational Linguistics.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2021. Self-training for unsupervised neural machine translation in unbalanced training data scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3975–3981, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Shaolei Wang, Zhongyuan Wang, Wanxiang Che, and Ting Liu. 2020. Combining self-training and self-supervised learning for unsupervised disfluency detection. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1813–1822, Online. Association for Computational Linguistics.

Yifan Wang, Jing Zhao, Junwei Bao, Chaoqun Duan, Youzheng Wu, and Xiaodong He. 2022. LUNA:

Learning slot-turn alignment for dialogue state tracking. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3319–3328, Seattle, United States. Association for Computational Linguistics.

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

Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. 2020a. 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, Steven C.H. Hoi, and Caiming Xiong. 2020b. Improving limited labeled dialogue state tracking with self-supervision. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4462–4472, Online. Association for Computational Linguistics.

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

Jiawei Wu, Xin Wang, and William Yang Wang. 2019b. Self-supervised dialogue learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3857–3867, Florence, Italy. Association for Computational Linguistics.

Qizhe Xie, Minh-Thang Luong, Eduard H. Hovy, and Quoc V. Le. 2020. Self-training with noisy student improves imagenet classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10684–10695. IEEE.

Puyang Xu and Qi Hu. 2018. An end-to-end approach for handling unknown slot values in dialogue state tracking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1448–1457, Melbourne, Australia. Association for Computational Linguistics.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. ConSERT: A contrastive framework for self-supervised sentence representation transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5065–5075, Online. 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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(16):14230–14238.

Fanghua Ye, Yue Feng, and Emine Yilmaz. 2022. AS-SIST: Towards label noise-robust dialogue state tracking. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2719–2731, Dublin, Ireland. Association for Computational Linguistics.

Fanghua Ye, Jarana Manotumruksa, Qiang Zhang, Shenghui Li, and Emine Yilmaz. 2021. Slot self-attentive dialogue state tracking. In *Proceedings of the Web Conference 2021*, pages 1598–1608.

Steve Young, Milica Gašić, Simon Keizer, François Mairesse, Jost Schatzmann, Blaise Thomson, and Kai Yu. 2010. The hidden information state model: A practical framework for pomdp-based spoken dialogue management. *Computer Speech & Language*, 24(2):150–174.

A Appendices

A.1 Experiment Results

Tables 7, 8, 9, 10 show all the experiments in joint goal accuracy of the four models: BASE, BASE w/ SSL, BASE w/ ST, CSS. Each of them is run on three random seeds for the four few-shot data ratio settings, and the final accuracy is averaged. Tables 11, 12 show the domain joint accuracy, and Tables 13, 14, 15, 16 show detailed experiments in domain joint accuracy on three different seeds.

Ratio	1-run	2-run	3-run	Average
1%	13.17	13.42	12.99	13.19
5%	35.82	37.73	38.02	37.19
10%	43.43	44.00	45.26	44.23
25%	50.46	48.19	48.95	49.20

Table 7: Joint goal accuracy of BASE.

Ratio	1-run	2-run	3-run	Average
1%	12.08	13.40	14.29	13.26
5%	36.50	39.71	39.97	38.73
10%	44.43	45.39	44.80	44.87
25%	49.12	50.12	49.21	49.48

Table 8: Joint goal accuracy of BASE w/ SSL.

Ratio	1-run	2-run	3-run	Average
1%	13.80	14.60	13.82	14.07
5%	40.74	40.23	40.01	40.33
10%	47.26	46.92	46.82	47.00
25%	51.87	51.66	51.81	51.78

Table 9: Joint goal accuracy of BASE w/ ST.

Ratio	1-run	2-run	3-run	Average
1%	12.35	14.52	15.32	14.06
5%	40.93	42.24	43.09	41.90
10%	47.87	48.63	47.39	47.96
25%	51.59	52.43	51.64	51.88

Table 10: Joint goal accuracy of CSS.

1%	attraction	hotel	restaurant	taxi	train	5%	attraction	hotel	restaurant	taxi	train
BASE	43.38	32.49	33.11	58.47	25.67	BASE	60.26	48.80	53.17	63.12	72.60
w/ SSL	43.35	32.60	31.34	58.49	25.70	w/ SSL	60.98	50.74	53.22	62.88	75.24
w/ ST	46.86	35.13	33.56	58.62	26.69	w/ ST	61.34	51.96	55.69	63.12	77.41
CSS	45.69	34.59	32.73	58.69	26.67	CSS	62.97	53.27	57.32	63.55	79.13

Table 11: Domain joint accuracy using 1% and 5% labeled training data.

10%	attraction	hotel	restaurant	taxi	train	25%	attraction	hotel	restaurant	taxi	train
BASE	66.82	54.50	59.97	69.33	77.03	BASE	69.49	57.26	66.38	78.00	79.21
w/ SSL	66.80	55.64	61.04	69.27	77.30	w/ SSL	71.13	57.91	66.92	78.60	78.47
w/ ST	68.56	56.57	62.46	69.85	80.12	w/ ST	71.57	59.67	66.88	78.13	81.38
CSS	68.14	57.43	64.25	70.52	79.97	CSS	71.83	58.01	68.41	78.43	80.89

Table 12: Domain joint accuracy using 10% and 25% labeled training data.

1%	attraction	hotel	restaurant	taxi	train
BASE	(42.76, 44.40, 42.98)	(35.92, 30.51, 31.04)	(31.38, 33.88, 34.06)	(58.19, 58.77, 58.45)	(24.65, 26.26, 26.11)
w/ SSL	(40.66, 48.05, 41.34)	(32.98, 32.39, 32.42)	(27.81, 32.99, 33.22)	(58.19, 58.13, 59.16)	(23.62, 25.34, 28.14)
w/ ST	(45.79, 49.56, 45.24)	(38.29, 32.67, 34.42)	(30.52, 34.00, 36.17)	(58.77, 58.65, 58.45)	(25.13, 27.85, 27.08)
CSS	(43.05, 49.63, 44.40)	(34.95, 35.32, 33.51)	(29.68, 34.80, 33.70)	(58.58, 58.52, 58.97)	(23.46, 26.16, 30.39)

Table 13: Domain joint accuracy using 1% labeled training data on three seeds.

5%	attraction	hotel	restaurant	taxi	train
BASE	(59.89, 60.86, 60.02)	(49.11, 49.11, 48.17)	(51.06, 54.51, 53.94)	(62.71, 62.52, 64.13)	(71.71, 71.27, 74.82)
w/ SSL	(60.31, 60.79, 61.83)	(50.64, 51.27, 50.30)	(51.15, 54.24, 54.27)	(61.03, 64.00, 63.61)	(72.25, 75.96, 77.52)
w/ ST	(62.99, 60.44, 60.60)	(53.74, 51.86, 50.27)	(53.89, 55.88, 57.31)	(62.39, 62.90, 64.06)	(77.31, 76.54, 78.40)
CSS	(62.79, 63.92, 62.21)	(55.39, 52.55, 51.86)	(55.73, 58.86, 57.37)	(62.53, 64.06, 64.06)	(79.08, 80.51, 77.81)

Table 14: Domain joint accuracy using 5% labeled training data on three seeds.

10%	attraction	hotel	restaurant	taxi	train
BASE	(64.80, 67.15, 68.51)	(55.49, 55.27, 52.74)	(59.42, 58.65, 61.83)	(70.52, 68.45, 69.03)	(76.89, 76.60, 77.60)
w/ SSL	(64.18, 67.67, 68.54)	(57.17, 56.24, 53.52)	(60.08, 61.18, 61.86)	(68.77, 69.29, 69.74)	(77.20, 76.86, 77.84)
w/ ST	(68.64, 67.34, 69.70)	(58.67, 55.33, 55.70)	(63.05, 62.55, 61.77)	(69.55, 69.35, 70.65)	(79.43, 79.77, 81.15)
CSS	(66.57, 68.86, 68.99)	(58.30, 56.86, 57.14)	(62.99, 66.15, 63.62)	(69.68, 70.52, 71.35)	(81.41, 79.43, 79.08)

Table 15: Domain joint accuracy using 10% labeled training data on three seeds.

25%	attraction	hotel	restaurant	taxi	train
BASE	(67.86, 69.44, 71.18)	(57.24, 57.49, 57.05)	(68.59, 64.78, 65.76)	(79.42, 76.84, 77.74)	(80.57, 78.10, 78.95)
w/ SSL	(69.54, 72.12, 71.73)	(58.39, 57.77, 57.58)	(66.63, 67.94, 66.18)	(78.58, 78.65, 78.58)	(79.19, 78.66, 77.55)
w/ ST	(71.44, 70.28, 72.99)	(58.77, 60.14, 60.11)	(66.89, 67.85, 65.91)	(77.94, 77.87, 78.58)	(82.03, 81.02, 81.10)
CSS	(71.67, 72.93, 70.89)	(56.92, 58.21, 58.89)	(67.52, 69.43, 68.29)	(78.39, 78.52, 78.39)	(81.89, 81.15, 79.64)

Table 16: Domain joint accuracy using 25% labeled training data on three seeds.