# Value type: the bridge to a better DST model

**Qixiang Gao**[1*], **Mingyang Sun**[1*]
**Yutao Mou**[1], **Chen Zeng**[1], **Weiran Xu**[1*]
[1]Beijing University of Posts and Telecommunications, Beijing, China
{gqx,mysun}@bupt.edu.cn
{myt,chenzeng,xuweiran}@bupt.edu.cn

## Abstract

Value type of the slots can provide lots of useful information for DST tasks. However, it has been ignored in most previous works. In this paper, we propose a new framework for DST task based on these value types. Firstly, we extract the type of token from each turn. Specifically, we divide the slots in the dataset into 9 categories according to the type of slot value, and then train a Ner model to extract the corresponding type-entity from each turn of conversation according to the token. Secondly, we improve the attention mode which is integrated into value type information between the slot and the conversation history to help each slot pay more attention to the turns that contain the same value type. Meanwhile, we introduce a sampling strategy to integrate these types into the attention formula, which decrease the error of Ner model. Finally, we conduct a comprehensive experiment on two multi-domain task-oriented conversation datasets, MultiWOZ 2.1 and MultiWOZ 2.4. The ablation experimental results show that our method is effective on both datasets, which verify the necessity of considering the type of slot value.

## 1 Introduction

Task-oriented dialogue systems have become more and more important as people's demand for life increases(booking flights or restaurants), which have become increasingly important in the field of NLP(Nature Language Process). (Henderson et al., 2019; Hung et al., 2021; Zheng et al., 2022) Traditionally, the task-oriented dialogue system consists of four modules (Zhang et al., 2020): Natural language understanding(NLU), Dialogue state tracking(DST), Dialogue manager(DM) and Natural language generation(NLG). This module directly affects the decision-making behavior of the dialogue system, and plays an extremely important role in the task-based dialogue system. (Lee et al., 2019)

**Sys**: yes, the autumn house is on the east part of time, the prices are cheap
and it is 4 stars. is there anything else you would like to know?
**Usr**: no, i just want to book it for 2 people for 5 nights starting wednesday.
**Turn_label**: hotel-area=east, hotel-book day=wednesday, hotel-people=2,
hotel-book stay=5, hotel-princerange=cheap, hotel-stars=4

**Sys**: your friend has good taste. It is located at 02:00 rose crescent city
centre, postcode cb23ll.
**Usr**: i would like to book a table for 7 people on Monday at 15:15 please.
**Turn_label**: restaurant-book day=monday, restaurant-book people=7,
restaurant-book time=15:15

**Sys**:
**Usr**: i would like a taxi from saint johns college to pizza hut fenditton.
**Turn_label**: taxi-departure=saint johns college,
taxi-destination=pizza hut fenditton

Figure 1: Common slot-value types in conversation, such as location, adjective, number and time.

The recent methods in DST work are mainly divided into two categories. The first category is based on ontology which means the candidate slot value is assumed to be known eg (Zhou et al., 2022; Ye et al., 2021b; Guo et al., 2021). The second is the way without ontology. These studies have completely abandoned ontology, and they assume that the slot value is unknown. eg(Wu et al., 2019; Kim et al., 2019; Kumar et al., 2020; Lin et al., 2021). However, most of their work is based on dialog state, dialog and slot modeling, ignoring that the value type of each slot may be different. If these slots are modeled uniformly, then there is a lack of a specific feature of each slot.

In this work, we propose a new DST framework named SVT-DST, which uses the **S**lot-**V**alue **T**ype as the bridge to increrase the model performance. With this method, each slot has specificity for the attention of the conversation history to better identify the slot value. Specifically, we first classify all the slots in the dataset according to their slot value types. As shown in Figure 1, adjectives, time and numbers correspond to pricerange, arrive-time and book-people respectively. We train a sequence annotation model with dialogue training which is used to extarct entities and corresponding entity-types in each on the turn. We hope that the attention

[*] The first two authors contribute equally. Weiran Xu is the corresponding author.

between the dialogue and slots can be higher when the turn is near to current turn with the same slot-value type. In order to achieve the goal, we use monotonically decreasing functions to integrate the attention weights, which will be described in detail in the method. we use monotonically decreasing functions to integrate these types into the attention operation.

Our main contributions are as follows: 1) We classify the slot according to the slot-value type, then train the Ner model to extract these types to improve the attention formula. 2)We design a sampling strategy to integrate these types into the attention formula, which decrease the error of Ner model. 3)We have achieved competitive results on MultiWOZ 2.1 and 2.4. We analyze the results and point out the future work.

# 2 Method

Figure 2 shows the structure of our DST model, including encoder, attention module and slot value processing module. In this section, we will introduce each module of this method in detail.

A T-turn conversation can be expressed as $C_t = \{(U_1, R_1), ..., (R_{t-1}, U_t)\}$, where $R_t$ represents system discourse and $U_t$ represents user discourse. We define the dialogue state of the t-th turn as $B_t = \{(S_j, V_j^t)| \quad 1 <= j <= J\}$, where $V_j^t$ represents the value of the j-th slot $S_j$ in the t-th turn. $J$ represents the number of predefined slots. Follow (Ren et al., 2018), we express the slot as a "domain slot" pair, such as 'restaurant-price range'.

## 2.1 Encoder

Follow(Ye et al., 2021b) , we use two bert (Devlin et al., 2018) models to encode context and slot respectively.

### 2.1.1 Context encoder

We express the dialogue at turn t as $D_t = R_t \oplus U_t$, where $\oplus$ represents sentence connection. Then the history of the dialogue including t-th turn as $M_t = D_1 \oplus D_2 \oplus ... \oplus D_t$.The input of the context encoder is $X_t = [CLS] \oplus M_t \oplus [SEP]$. The output of the encoder is:

$$C_t = bert_{finetuned}(X_t) \tag{1}$$

Where $C_t \in R^{|X_t| \times d}$, $|X_t|$ is the length of $M_t$ and $d$ is the hidden size of bert. $bert_{finetuned}$ indicates that the bert model updates a part of parameters during training.

### 2.1.2 Slot-value related encoder

We employ the first token to represent the aggregate representation of the entire input sequence. Therefore, for any slot $S_j \in S(1 \leq j \leq J)$ and any value $v_j^t \in V_j$ we have:

$$h^{S_j} = bert_{fixed}(S_j) \in R^{1 \times d} \tag{2}$$

$$h^{v_j^t} = bert_{fixed}(v_j^t) \in R^{1 \times d} \tag{3}$$

For the last turn of dialogue state $B_{t-1}$, we have

$$h^{B_{t-1}} = bert_{fixed}(B_{t-1}) \tag{4}$$

Where $h^{B_{t-1}} \in R^{|B_{t-1}| \times d}$, $B_{-1} = Null$. $bert_{fixed}$ indicates that the bert model has fixed parameters during training.

## 2.2 Cross-Attention

We use the multi-head-attention module(Vaswani et al., 2017) as the basis of our attention module.

### 2.2.1 Slot-Context Attention

We first calculate the bias term of the attention formula. For each dialogue history $M_t$, we first use the monotonically decreasing distribution function $\eta(n)$ to initialize the weight of each turn of dialogue $D_t$ in the dialogue history:

$$\psi(n) = \int_n^{n+1} \eta(n) dn \tag{5}$$

Where $n = T - t$, $n$ represents the distance between the last turn and the current turn. The closer the distance is, the greater the weight will be obtained. Note that $\psi(T)$ represents the weight of distance T for this turn (turn 0) and the latest turn t. We record the turns of the value type $type_j$ with slot $S_j$ in the history:

$$\omega = [m, ..., n] \tag{6}$$

Where n>m, which represents the turn indexs. Then we calculate the weight of these turns:

$$\Omega_{j,t}^i = \begin{cases} \psi(T-i), i \in \omega \\ 0, else \end{cases} \tag{7}$$

Finally, we add these two weights according to the turn indexs to get bias:

$$bias_{j,t} = \Omega_{j,t} = [\Omega_{j,t}^0, ..., \Omega_{j,t}^t] \tag{8}$$

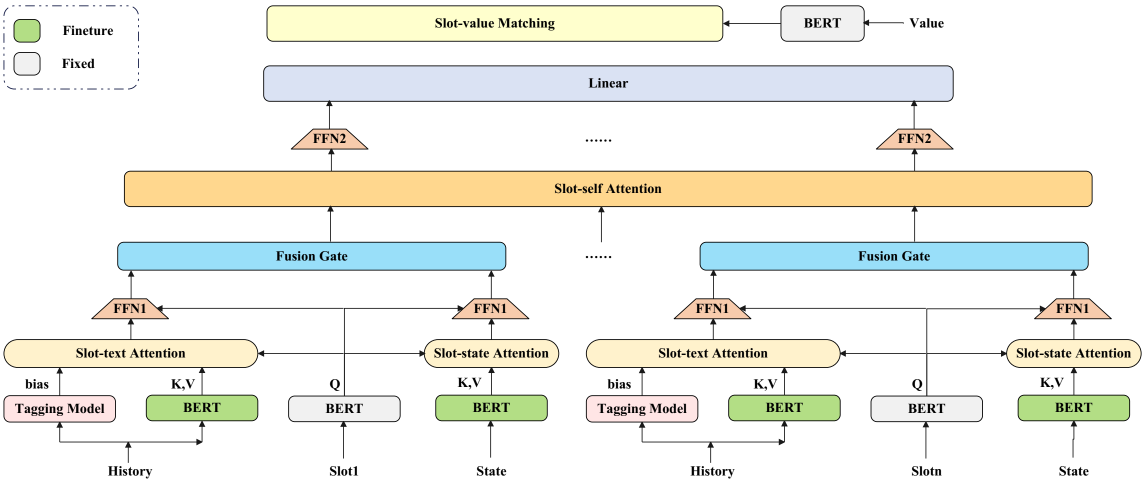

Figure 2: The overall architecture of our proposed Model.

The attention between $S_j$ and $C_t$ can be calculated as:

$$A^C_{j,t} = Softmax(\frac{Q_j K^T_t}{\sqrt{d_k}} + \varphi(bias_{j,t}) W_{bias}) V_t \tag{9}$$

$$A^{C,FFN}_{j,t} = W^r_2 ReLU(W^r_1[(h^{S_j}, A^C_{j,t}] + b^r_1) + b^r_2 \tag{10}$$

Where $\varphi()$ indicates a learnable mapping built by embedding. $W_{bias}$, $W^r_1$ and $W^r_2$ indicates a linear layer, respectively.

#### 2.2.2 Slot-State Attention

For $S_j$ and $B_{t-1}$, their attention can be expressed as:

$$A^B_{j,t-1} = MultiHead(h^{S_j}, h_{B_{t-1}}, h_{B_{t-1}}) \tag{11}$$

$$\begin{aligned} A^{B,FFN}_{j,t-1} =& W^r_4 ReLU(W^r_3[(h^{S_j}, A^B_{j,t-1}] + b^r_1) \\ &+ b^r_2 \end{aligned} \tag{12}$$

#### 2.2.3 Gate Fusion

inspired by (Zhou et al., 2022), we employ a gate module to combine the attention between Slot-context and Slot-state:

$$g^t_j = \sigma(W^s_j \otimes [A^{C,FFN}_{j,t}; A^{B,FFN}_{j,t-1}]) \tag{13}$$

$$m^t_j = g^t_j \cdot A^C_{j,t} + (1 - g^t_j) \cdot A^B_{j,t-1} \tag{14}$$

Where $\otimes$ indicates vector product, $\sigma$ indicates the sigmoid function and $\cdot$ indicates element-wise product operation.

### 2.3 Self-Attention And Value Matching

In this part, we have followed the relevant part of (Ye et al., 2021b).

### 2.4 Ner Model And Sampling Strategy

We employ the W2NER model(Li et al., 2022) as our tagging model. The strategy of our label-making is that: for each value in the ontology, if the value is in current turn, we will tagging this value. For sampling strategy, only when the target entities are different from entities extracted from previous turns, this turn will be marked with the entities' type. This strategy helps to reduce the interference of duplicate entities. For the specific classification of each slot, please refer to the appendix. In particular, for bool type, we train the annotation model to extract keywords, such as internet, parking, etc.

### 2.5 Optimization

We use the sum of the negative log-likelihood as the loss function at each turn $t$:

$$L_t = -\sum_{j=1}^{J} log(P(V^t_j \mid X_t, S_t)) \tag{15}$$

Where

$$P(V^t_j \mid X_t, S_t) = \frac{exp(-||\gamma^t_{S^t_j} - h^{V^t_j}||_2)}{\sum\limits_{V'_j \in V_j} exp(-||\gamma^t_{S^t_j} - h^{V'_j}||_2)} \tag{16}$$

$\gamma^t_{S^t_j}$ indicates the output of self-attention module corresponding to $S_j$ at the t-th turn.

| Model | Joint Goal Acc | |
|---|---|---|
| | 2.1 | 2.4 |
| Trade | 45.60% | 55.05% |
| Tripy | 55.18% | 64.75% |
| MinTL-BART | 53.62% | - |
| STAR | 56.36% | 73.62% |
| MSP-B | 56.20% | - |
| Tripy-R | 55.99% | 69.87% |
| SST | 55.23% | - |
| LUNA | 57.62% | - |
| Frame-Base | 53.28% | 66.15% |
| Ours(NER) | 55.37% | 68.93% |
| Ours(NER wo:SP) | 53.68% | 66.46% |
| Ours(GD) | 59.27% | 75.01% |

Table 1: Main results on MultiWOZ 2.1 and 2.4 datasets. NER, wo:SP and GD mean that train the model without sampling strategy and train the model with ground truth slot-value types, respectively.

# 3 Experiments

## 3.1 Dataset, metric and Evaluation

We evaluate our method on these datasets: MultiWOZ 2.1 (Eric et al., 2019) and MultiWOZ 2.4 (Ye et al., 2021a) which provide turn-level annotations of dialogue states in 7 different domains. We evaluate our method on this dataset and follow the pre-processing and evaluation setup from (Wu et al., 2019), where restaurant, train, attraction, hotel, and taxi domains are used for training and testing. We use Joint Goal Accuracy that is the average accuracy of predicting all slot assignments for a given service in a turn correctly to evaluate the main results of models.

## 3.2 Baselines

(1) Trade: Transferable dialogue state generator (Wu et al., 2019) which utilizes copy mechanism to facilitate domain knowledge transfer. (2) Tripy: It applies three copying mechanisms to extract all values (Heck et al., 2020) (3) MinTL: An effective transfer learning framework for task-oriented dialogue systems(Lin et al., 2020),which uses T5 (Raffel et al., 2020) and Bart (Lewis et al., 2019). (4) Star: Framework with self-attention modules to learn the relationship between slots better (Ye et al., 2021b) from the dialogue context. (5) SST: a multi-domain dialogue state tracker which employs graph methods to fuse utterance and schema graph.(Chen et al., 2020) (6) TripyR: The model with a new training strategy based on Tripy (Heck et al., 2022). (7) MSP-B: An extraction model with mentioned slot pool(MSP) (Sun et al., 2022) (8) LUNA: It applies a slot-turn alignment strategy to accurately locate slot values and their associated context. (Wang et al., 2022)

| Function | Joint Goal Acc | |
|---|---|---|
| | 2.1 | 2.4 |
| y = 1/2*(x+1) | 53.79% | 68.06% |
| y = 1/(x+1) | 53.80% | 68.06% |
| y = (x-30)^2/900 | 54.79% | 67.47% |
| y = 1/2*(x+1)+1 | 55.08% | 69.75% |
| y = 1-x/30 | 55.37% | 68.93% |

Table 2: Results of different functions on MultiWOZ 2.1 and 2.4 datasets. $y = 1 - x/30$ is used in the main experiments.

## 3.3 Main Results And Analysis Experiments

Table 1 shows the results of our main test and ablation study. Our base model achieved 53.28% for the joint-acc, while our Ner-based model achieved 55.37% , a significant improvement of 2.09% compared with the base model. In 2.4 dataset, our model achieved 68.28%, a significant improvement of 2.93% compared with the base model. And When we use the correct type labels for training, the model performance reaches 59.27%, which has exceeded all baseline models. Ground truth is extracted according to the slot-type in the turn label, similar to our sampling strategy. In order to model the attention of state and dialog history separately, we changed the attention in Star(Ye et al., 2021b) to the fusion of slot attention and dialog history attention. Such changes reduced the performance of the model. However, the ablation experiment shows that the method we proposed can really benefit the model indicators.

Table 2 shows the results of our analysis experiments, which use different distribution functions to model attention. For both 2.1 and 2.4 datasets, the experimental results show that under different distribution function modeling, the distribution with constant term bias may produce higher results such as $0.5 * (1 + x) + 1$ and $1 - x/30$. And it often has a positive impact on the experiment when the power of the independent variable is 1.

## 3.4 Case Study

We conducted a series of analytical experiments on attention weights. As shown in the Table 3, we randomly selected a slot, "attraction-name," and then chose an example PMUL4648 from the test set to observe the attention distribution of this slot

|  |  | Attention Score |  |  |  |  |
|---|---|---|---|---|---|---|
| Function | param\turn | 0 | 1 | 2 | 3 | 4 |
| y = 1/2*(x+1) | score | 2.6213 | 1.9166 | 2.9988 | 2.9706 | 0.3718 |
|  | b | -0.2587 | -0.1632 | -0.2587 | -0.2587 | -0.2521 |
|  | score+b | 2.3626 | 1.7534 | 2.7401 | 2.7120 | 0.1196 |
| y = (x-30)^2/900 | score | 5.5183 | 2.5206 | 2.6990 | 2.0383 | -0.1586 |
|  | b | 0.1107 | -0.1631 | 0.1107 | 0.1107 | 0.0921 |
|  | score+b | 5.6291 | 2.3576 | 2.8097 | 2.1490 | -0.0666 |
| y = 1/2*(x+1)+1 | score | 4.3446 | 2.7369 | 3.2936 | 3.4940 | 0.3512 |
|  | b | -0.2793 | -0.1633 | -0.2793 | -0.2793 | -0.2714 |
|  | socre+b | 4.0653 | 2.5737 | 3.0143 | 3.2146 | 0.0798 |

Table 3: One case of the attention between the attraction-name slot and context for dialogue PMUL4648 in the 2.4 dataset. Score denotes $QK/\sqrt{dk}$ and b denotes the attention bias

for each turn in the test samples. In the example, the attraction-name slot is activated in the turn 2. It can be seen that function 3 noticed this turn with a large weight, followed by function 1. As a comparison, function 2 assigned larger weights to the first turn, which is sufficient to indicate that the fitting effect of function 2 is weaker compared to the other two functions. Our analysis is as follows: If there is no constant term in the distribution function, the difference between score+bias and score is not significant, resulting in limited performance improvement of the model. On the other hand, the power of the independent variable is greater than 1 such as function 2, the magnitude changes too obviously after Softmax. This leads to not smooth transitions between turns, resulting in limited performance improvement.

The result of using the ground truth labels training model shows that there is still huge space for improvement in Ner model annotation. One of the biggest challenges is that the annotation model often assigns certain entities to labels based on some fragmented tokens, without considering the impact of context, which leads to the proliferation of labels. We will solve this problem in future work.

## 4 Conclusion

In this paper, we propose an effective method to integrate slot-types into the DST model. Specifically, we propose the SVT-DST. This framework incorporates the slot-types information into the attention operation to help model pay more attention to these turns that include the type of one slot. Further, We design a sampling strategy to integrate these types into the attention formula to decrease the error of Ner model. Results on MultiWOZ dataset show that our method has significant improvement on this task.

## Limitation

This work has two main limitations: (1) The performance of the model largely depends on the performance of the annotation model. If the annotation model is too simple, it may cause the performance of the DST model to decline. On the contrary, it will increase the complexity of the overall model and prolong the reasoning time. (2) Even for the labeling model with good performance, the tagging values may also interfere with the DST model. For details, please refer to the analysis experiment.

## References

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 *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7521–7528.

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

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, and Dilek Hakkani-Tür. 2019. Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines.

Jinyu Guo, Kai Shuang, Jijie Li, and Zihan Wang. 2021. Dual slot selector via local reliability verification for dialogue state tracking. *arXiv preprint arXiv:2107.12578*.

Michael Heck, Nurul Lubis, Carel van Niekerk, Shutong Feng, Christian Geishauser, Hsien-Chin Lin, and Milica Gašić. 2022. Robust dialogue state tracking with weak supervision and sparse data. *Transactions of the*

*Association for Computational Linguistics*, 10:1175–1192.

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

Matthew Henderson, Ivan Vulić, Daniela Gerz, Iñigo Casanueva, Paweł Budzianowski, Sam Coope, Georgios Spithourakis, Tsung-Hsien Wen, Nikola Mrkšić, and Pei-Hao Su. 2019. Training neural response selection for task-oriented dialogue systems. *arXiv preprint arXiv:1906.01543*.

Chia-Chien Hung, Anne Lauscher, Simone Paolo Ponzetto, and Goran Glavaš. 2021. Ds-tod: Efficient domain specialization for task oriented dialog. *arXiv preprint arXiv:2110.08395*.

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

Adarsh Kumar, Peter Ku, Anuj Goyal, Angeliki Metallinou, and Dilek Hakkani-Tur. 2020. Ma-dst: Multi-attention-based scalable dialog state tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8107–8114.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019. Sumbt: Slot-utterance matching for universal and scalable belief tracking. *arXiv preprint arXiv:1907.07421*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, and Fei Li. 2022. Unified named entity recognition as word-word relation classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10965–10973.

Zhaojiang Lin, Bing Liu, Seungwhan Moon, Paul Crook, Zhenpeng Zhou, Zhiguang Wang, Zhou Yu, Andrea Madotto, Eunjoon Cho, and Rajen Subba. 2021. Leveraging slot descriptions for zero-shot cross-domain dialogue state tracking. *arXiv preprint arXiv:2105.04222*.

Zhaojiang Lin, Andrea Madotto, Genta Indra Winata, and Pascale Fung. 2020. Mintl: Minimalist transfer learning for task-oriented dialogue systems. *arXiv preprint arXiv:2009.12005*.

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.

Liliang Ren, Kaige Xie, Lu Chen, and Kai Yu. 2018. Towards universal dialogue state tracking. *arXiv preprint arXiv:1810.09587*.

Zhoujian Sun, Zhengxing Huang, and Nai Ding. 2022. On tracking dialogue state by inheriting slot values in mentioned slot pools. *arXiv preprint arXiv:2202.07156*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yifan Wang, Jing Zhao, Junwei Bao, Chaoqun Duan, Youzheng Wu, and Xiaodong He. 2022. Luna: Learning slot-turn alignment for dialogue state tracking. *arXiv preprint arXiv:2205.02550*.

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. *arXiv preprint arXiv:1905.08743*.

Fanghua Ye, Jarana Manotumruksa, and Emine Yilmaz. 2021a. Multiwoz 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections to improve state tracking evaluation. *arXiv preprint arXiv:2104.00773*.

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

Zheng Zhang, Ryuichi Takanobu, Qi Zhu, MinLie Huang, and XiaoYan Zhu. 2020. Recent advances and challenges in task-oriented dialog systems. *Science China Technological Sciences*, 63(10):2011–2027.

Xuanzhi Zheng, Guoshuai Zhao, Li Zhu, Jihua Zhu, and Xueming Qian. 2022. What you like, what i am: Online dating recommendation via matching individual preferences with features. *IEEE Transactions on Knowledge and Data Engineering*.

Yihao Zhou, Guoshuai Zhao, and Xueming Qian. 2022. Dialogue state tracking based on hierarchical slot attention and contrastive learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4737–4741.

# A Appendix

## A.1 Slot-Value Type

## A.2 Implementation

we implement SVT-DST model based on the bert-base-uncased (110M parameters) model which has

| slot | type |
|---|---|
| xxx-name | location |
| xxx-departure | location |
| xxx-destination | location |
| xxx-area | area |
| xxx-day | day |
| xxx-type | type |
| xxx-stay | number |
| xxx-book people | number |
| xxx-stars | number |
| xxx-arriveby | time |
| xxx-leaveat | time |
| restaurant-food | food |
| xxx-pricerange | adjective |
| hotel-parking | bool |
| hotel-internet | bool |

Table 4: Type classification corresponding to each slot.

12 layers and the hidden size is 768. The quantity of trainable parameters of the whole model is 24.85M. Our model is trained with a base learning rate of 0.0001 for 12 epochs about 4 hours. We use 1 NVIDIA 3090 GPU for all of our experiments.Joint goal accuracy is used to evaluate the performance of the models. Predicted dialogue states are correct only when all of the predicted values exactly match the correct values.The result of the model comes from the result of two averages. The annotation model is based on w2ner, which uses bert-large-cased (330M parameters) as encoder.

## ACL 2023 Responsible NLP Checklist

### A For every submission:

☑ A1. Did you describe the limitations of your work?
*5.Limitation*

☒ A2. Did you discuss any potential risks of your work?
*Our work is devoted to improving the quality of the dialogue state tracking model, and there is basically no potential risk in theory.*

☑ A3. Do the abstract and introduction summarize the paper's main claims?
*Abstract and Section 1*

☒ A4. Have you used AI writing assistants when working on this paper?
*Left blank.*

### B ☒ Did you use or create scientific artifacts?

*Left blank.*

□ B1. Did you cite the creators of artifacts you used?
*No response.*

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
*No response.*

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
*No response.*

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
*No response.*

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
*No response.*

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
*No response.*

### C ☑ Did you run computational experiments?

*3*

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
*Appendix*

*The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.*

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
*Section 3.3*

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
*Appendix*

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
*Appendix*

**D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?**

*Left blank.*

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
*No response.*

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
*No response.*

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
*No response.*

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
*No response.*

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
*No response.*