

Dialogue State Tracking with Multi-Level Fusion of Predicted Dialogue States and Conversations

Jingyao Zhou^{1,2} Haipang Wu^{1,2} Zehao Lin³ Guodun Li³ Yin Zhang^{3*}

¹Hithink RoyalFlush Information Network Co., Ltd

²Hithink RoyalFlush AI Research Institute

³College of Computer Science and Technology, Zhejiang University

{zhoujingyao, wuhaipang}@myhexin.com

{georgenlin, guodun.li, zhangyin98}@zju.edu.cn

Abstract

Most recently proposed approaches in dialogue state tracking (DST) leverage the context and the last dialogue states to track current dialogue states, which are often slot-value pairs. Although the context contains the complete dialogue information, the information is usually indirect and even requires reasoning to obtain. The information in the lastly predicted dialogue states is direct, but when there is a prediction error, the dialogue information from this source will be incomplete or erroneous. In this paper, we propose the Dialogue State Tracking with Multi-Level Fusion of Predicted Dialogue States and Conversations network (FPDSC). This model extracts information of each dialogue turn by modeling interactions among each turn utterance, the corresponding last dialogue states, and dialogue slots. Then the representation of each dialogue turn is aggregated by a hierarchical structure to form the passage information, which is utilized in the current turn of DST. Experimental results validate the effectiveness of the fusion network with 55.03% and 59.07% joint accuracy on MultiWOZ 2.0 and MultiWOZ 2.1 datasets, which reaches the state-of-the-art performance. Furthermore, we conduct the deleted-value and related-slot experiments on MultiWOZ 2.1 to evaluate our model.

1 Introduction

Dialogue State Tracking (DST) is utilized by the dialogue system to track dialogue-related constraints and user's requests in the dialogue context. Traditional dialogue state tracking models combine semantics extracted by language understanding modules to estimate the current dialogue states (Williams and Young, 2007; Thomson and Young, 2010; Wang and Lemon, 2013; Williams, 2014), or to jointly learn speech understanding (Henderson et al., 2014; Zilka and Jurcicek, 2015; Wen

et al., 2017). They rely on hand-crafted features and complex domain-specific lexicons, which are vulnerable to linguistic variations and difficult to scale. Recently proposed approaches attempt to automatically learn features from the dialogue context and the previous dialogue states. Most of them utilize only the context (Shan et al., 2020) or encode the concatenation of context and dialogue states (Hosseini-Asl et al., 2020) or utilize a simple attention mechanism to merge the information from the above two sources (Ouyang et al., 2020). These methods do not fully exploit the nature of the information in the context and the predicted dialogue states. The information nature of the context is complete and may be indirect. While the nature of the predicted dialogue states is direct and may be incomplete or erroneous.

Our FPDSC model exploits the interaction among the turn utterance, the corresponding last dialogue states, and dialogue slots at each turn. A fusion gate (the turn-level fusion gate) is trained to balance the keep-proportion of the slot-related information from the turn utterance and the corresponding last dialogue states at each turn. Then it applies a hierarchical structure to keep the complete information of all dialogue turns. On top of the model, we employ another fusion gate (the passage-level fusion gate) to strengthen the impact of the last dialogue states. Ouyang et al. (2020) shows that such strengthening is vital to solve the related-slot problem. The problem is explained in Table 1. To eliminate the negative impact of the error in the predicted dialogue states, we train our models in two phases. In the teacher-forcing phase, previous dialogue states are all true labels. While in the uniform scheduled sampling phase (Bengio et al., 2015), previous dialogue states are half predicted dialogue states and half true labels. Training with such natural data noise from the error in the predicted dialogue states helps improve the model's

*Corresponding Author

U_1 : I need a place to dine in the centre.
State: restaurant-area=centre
S_2 : I recommend the rice house. Would you like me to reserve a table?
U_2 : Yes, please book me a table for 9.
State: restaurant-area=centre; restaurant-book people=9; restaurant-name=rice house
S_3 : Unfortunately, I could not book the rice house for that amount of people.
U_3 : please find another restaurant for that amount of people at that time.
State: restaurant-area=centre; restaurant-book people=9 restaurant-name=none
S_4 : how about tang restaurant ?
U_4 : Yes, please make me a reservation. I also need a taxi.
State: restaurant-area=centre; restaurant-book people=9 restaurant-name=tang;
S_5 : What is your destination ?
U_5 : To the restaurant.
State: restaurant-area=centre; restaurant-book people=9 restaurant-name=tang; taxi-destination=tang

Table 1: An example of dialogue contains (1) the *deleted-value problem* at the 3rd turn, which changes restaurant-name from rice house to none, and (2) the *related-slot phenomenon* at the 5th turn, which carries over the value from restaurant-name to taxi-destination.

robustness.

We design an ablation study for FPDSC, the variants of which are as follows: base model (without turn/passage-level fusion gates), turn-level model (with only turn-level fusion gate), passage-level model (with only passage-level fusion gate) and dual-level model (with both turn/passage-level fusion gates). We also design the experiment for the deleted-value problem, which is explained in Table 1, and the related-slot problem. Besides, we design two comparative networks to validate the effectiveness of the turn-level fusion gate and the whole previous dialogue states. One comparative network employs only the attention mechanism to merge information from the turn utterance, the corresponding last dialogue states, and dialogue slots at each turn. Another comparative network utilize only the last previous dialogue states in the turn-level fusion gate. Our model shows strong performance on MultiWOZ 2.0 (Budzianowski et al., 2018) and MultiWOZ 2.1 (Eric et al., 2019) datasets. Our main contributions are as follows:

- We propose a novel model, which utilizes multi-level fusion gates and the attention mechanism to extract the slot-related information from the conversation and previous dialogue states. The experimental results of two comparative networks validate the effec-

tiveness of the turn-level fusion gate to merge information and the importance of the whole previous dialogue states to improve DST performance.

- Both turn/passage-level fusion between the context and the last dialogue states helps at improving the model’s inference ability. The passage-level fusion gate on the top of the model is more efficient than the turn-level fusion gate on the root for slot correlation problem. While the turn-level fusion gate is sensitive to signal tokens in the utterance, which helps improve the general DST performance.
- Experimental results on the deleted-value and the related-slot experiment shows the ability of the structure to retrieve information. Besides, our models reach state-of-the-art performance on MultiWOZ 2.0/2.1 datasets.

2 Related Work

Recently proposed methods show promising progress in the challenge of DST. CHAN (Shan et al., 2020) employs a contextual hierarchical attention network, which extracts slot attention based representation from the context in both token- and utterance-level. Benefiting from the hierarchical structure, CHAN can effectively keep the whole dialogue contextual information. Although CHAN achieves the new state-of-the-art performance on MultiWoz 2.0/2.1 datasets, it ignores the information from the predicted dialogue states. Figures 1 and 2 show the difference between CHAN and FPDSC in the extraction of the slot-related information in one dialogue turn.

In the work of Ouyang et al. (2020), the problem of slot correlations across different domains is defined as related-slot problem. DST-SC (Ouyang et al., 2020) model is proposed. In the approach, the last dialogue states are vital to solve the related-slot problem. The method merges slot-utterance attention result and the last dialogue states with an attention mechanism. However, the general performance of DST-SC is worse than CHAN.

SOM-DST (Kim et al., 2020) and CSFN-DST Zhu et al. (2020) utilize part of the context and the last dialogue states as information sources. The two methods are based on the assumption of Markov property in dialogues. They regard the last dialogue states as a compact representation of the whole dialogue history. Once a false prediction of a slot

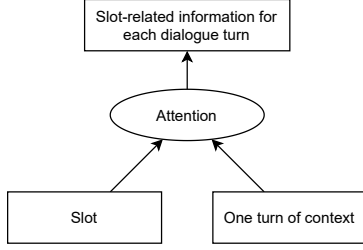


Figure 1: A part structure of CHAN and FPDSC (base/passage-level)

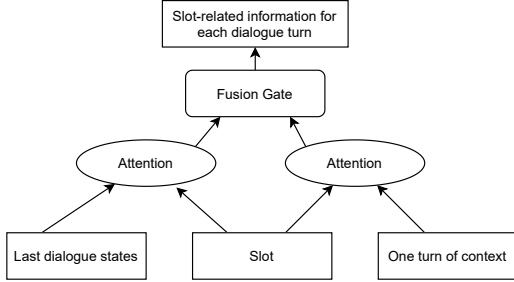


Figure 2: A part structure of FPDSC (turn/dual-level)

exists and the slot-related context is dropped, the dialogue states will keep the error.

3 Model

Figure 3 shows the overall structure of FPDSC (dual-level). The followings are important notations for our model.

Inputs: The context $D = \{(U_1, R_1), \dots, (U_t, R_t)\}$ where U_t and R_t represent utterance for user and system at the t -th dialogue turn; The previous dialogue states $B = \{B_1, \dots, B_{t-1}\}$ where $B_t = \{(s, v_t), s \in \mathcal{S}\}$, \mathcal{S} is slot set, s is one of the slot names, v_t is the corresponding slot value at the t -th turn; $\mathcal{V} = \{V_s, s \in \mathcal{S}\}$ are slot value candidates of all slots .

Turn-level Information: The slot-related information for each dialogue turn in Figure 2 is the turn-level information. In Figure 3, the turn-level information is denoted as $\{m_1^{s,tl}, \dots, m_{t-1}^{s,tl}, m_t^{s,tl}\}$, which is the fusion (the turn-level fusion gate) result of the slot-utterance attention results $\{c_1^s, \dots, c_{t-1}^s, c_t^s\}$ and the slot-dialogue-states attention results $\{l_1^s, \dots, l_{t-1}^s, l_t^s\}$. The weights $\{g_1^{s,tl}, \dots, g_{t-1}^{s,tl}, g_t^{s,tl}\}$ are from the same fusion gate, which is utilized to allocate the keep-proportion from the conversations and previous dialogue states. The turn-level information of a slot is fed to a transformer encoder to form the **mutual interaction information** $\{h_{t,1}^{s,tl}, \dots, h_{t,t-1}^{s,tl}, h_{t,t}^{s,tl}\}$.

Passage-level Information: The attention $\{\text{Attention}_3\}$ result of the **mutual interaction information** and a slot is the passage-level information $\{m_t^{s,pl}\}$ of a slot.

Core Feature: The weight $\{g_t^{s,pl}\}$ are applied to balance the turn-level information of the current dialogue turn $\{m_t^{s,tl}\}$ and the passage-level information $\{m_t^{s,pl}\}$ of a slot. We employ the attention $\{\text{Attention}_4\}$ mechanism between the turn/passage-level balanced information $\{f_t^{s,pl}\}$ and the last dialogue states $\{h_t^l\}$ to strengthen the impact of the last dialogue states. Another weight $\{g_t^{s,pl'}\}$ (from the passage-level fusion gate) merge the turn/passage-level balanced information $\{f_t^{s,pl}\}$ and the strengthened information $\{f_t^{s,pl'}\}$ to form the core feature $\{f_t^s\}$, which is utilized in the downstream tasks.

3.1 BERT-Base Encoder

Due to pre-trained models' (e.g., BERT) strong language understanding capabilities (Mehri et al., 2020), we use the fixed-parameter BERT-Base encoder (BERT_{fixed}) to extract the representation of slot names, slot values and the previous dialogue states. Three parts share the same parameters from HuggingFace¹. We also apply a tunable BERT-Base encoder (BERT_{tunable}) to learn the informal and noisy utterances distribution (Zhang et al., 2020b) in the dialogue context. The two BERT-Base Encoders are input layers of the model. [CLS] and [SEP] represent the beginning and the end of a text sequence. We use the output at [CLS] to represent the whole text for BERT_{fixed}. A slot-value pair in the last dialogue states at the t -th turn is denoted as:

$$h_t^{ls} = \text{BERT}_{fixed}([s; v_{t-1}]) \quad (1)$$

where h_t^{ls} is the s slot-related representation of last dialogue state at the dialogue t -th turn. Thus the full representation of the last dialogue states at the t -th turn is as follows:

$$h_t^l = h_t^{ls_1} \oplus \dots \oplus h_t^{ls_k} \dots \oplus h_t^{ls_n} \quad l_{s_k} \in \mathcal{S} \quad (2)$$

\oplus means concatenation. The entire history of the dialogue states is $H_t^p = \{h_1^l, \dots, h_{t-1}^l, h_t^l\}$. The representations of slot s and its corresponding value v_t are as follows:

$$h^s = \text{BERT}_{fixed}(s) \quad (3)$$

¹<https://huggingface.co/>

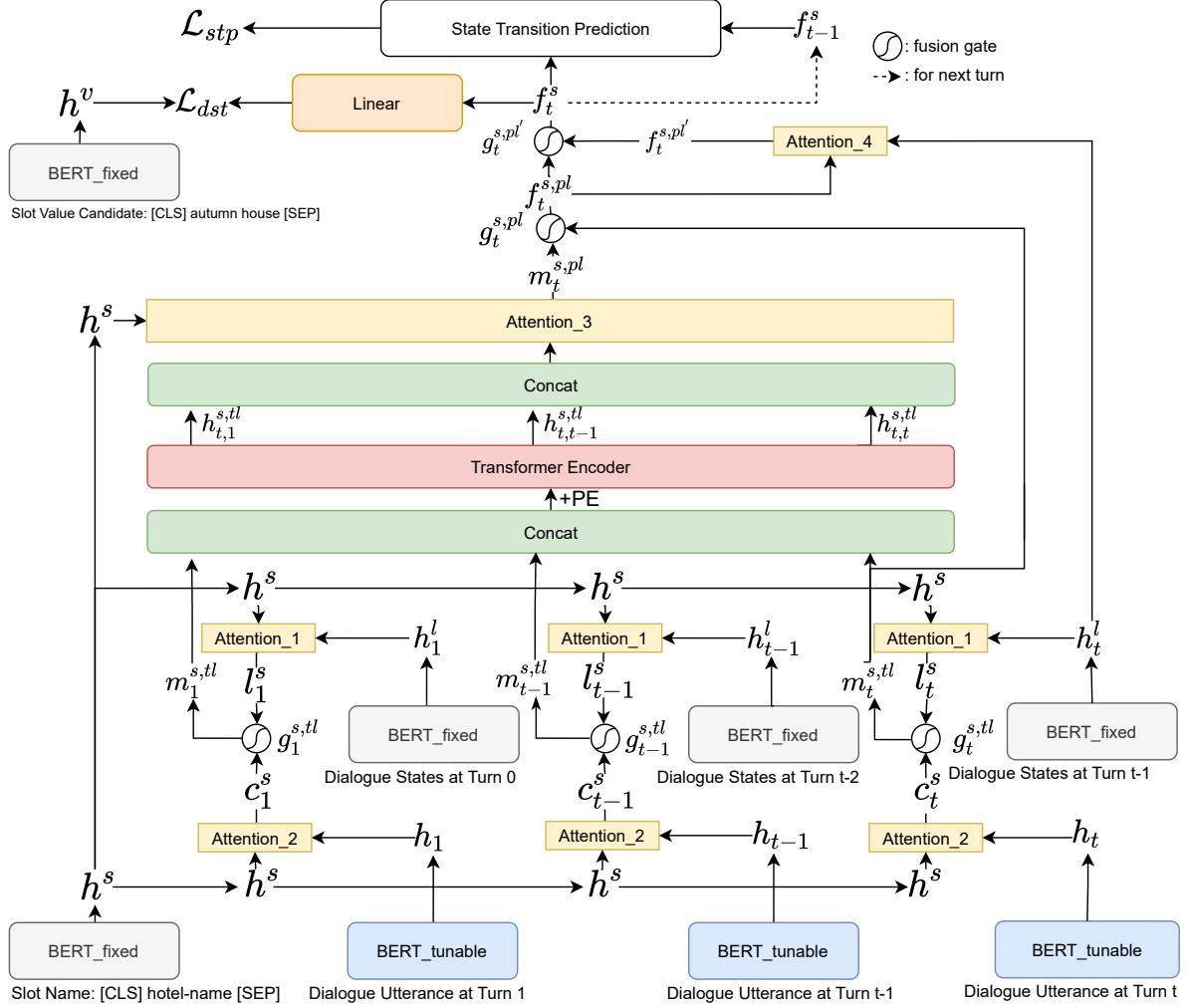


Figure 3: The structure of **dual-level** FPDS. The dialogue utterance at the t -th turn is $[\text{CLS}]R_t[\text{SEP}]U_t[\text{SEP}]$. The dialogue state is a list of slot-value pairs ($[\text{CLS}]\text{Slot}_1[\text{SEP}]\text{Value}_1[\text{SEP}], \dots, [\text{CLS}]\text{Slot}_n[\text{SEP}]\text{Value}_n[\text{SEP}]$). All slot values are none in the initial dialogue states. The **turn-level** approach is without Attention_4 and the passage-level fusion gate ($g_t^{s,pl'}$ is the output weight of the gate). The **passage-level** approach is without Attention_1 and the turn-level fusion gate ($\{g_1^{s,tl}, \dots, g_{t-1}^{s,tl}, g_t^{s,tl}\}$ are the output weights of the gate). The **base** approach is without Attention_1/4 and turn/passage-level fusion gate. The **base** approach has the same structure as CHAN with different early stop mechanism.

$$h^v = \text{BERT}_{fixed}(v_t) \quad (4) \quad \text{are as follows:}$$

$\text{BERT}_{tunable}$ extracts utterances distribution of user $U_t = \{w_1^u, \dots, w_l^u\}$ and system $R_t = \{w_1^r, \dots, w_l^r\}$ at the t -th turn, which are marked as:

$$h_t = \text{BERT}_{tunable}([R_t; U_t]) \quad (5)$$

The dialogue context until t -th turn is $H_t = \{h_1, \dots, h_{t-1}, h_t\}$.

3.2 MultiHead-Attention Unit

We utilize MultiHead-Attention (Vaswani et al., 2017) here to get the slot-related information from the turn utterance and the corresponding last dialogue states. The representations at the t -th turn

$$l_t^s = \text{Attention}_1(h^s, h_t^l, h_t^l) \quad (6)$$

$$c_t^s = \text{Attention}_2(h^s, h_t, h_t) \quad (7)$$

Another attention unit is applied to get the passage-level information of a slot from the mutual interaction information $H_t^{s,tl} = \{h_{t,1}^{s,tl}, \dots, h_{t,t-1}^{s,tl}, h_{t,t}^{s,tl}\}$, which is described in section 3.3.

$$m_t^{s,pl} = \text{Attention}_3(h^s, H_t^{s,tl}, H_t^{s,tl}) \quad (8)$$

We apply an attention unit to connect the representation of the merged turn/passage-level balanced information $f_t^{s,pl}$ and the last dialogue states to

enhance the impact of the last dialogue states.

$$f_t^{s,pl'} = \text{Attention}_4(f_t^{s,pl}, h_t^l, h_t^l) \quad (9)$$

$f_t^{s,pl'}$ is the enhanced result. All attention units above do not share parameters.

3.3 Transformer Encoder

The complete turn-level merged information $M_t^{s,tl} = \{m_1^{s,tl}, \dots, m_{t-1}^{s,tl}, m_t^{s,tl}\}$ has no dialogue sequence information. Besides, each turn representation does not fully share information. Thus we apply a transformer encoder (Vaswani et al., 2017).

$$H_t^{s,tl} = \text{TransformerEncoder}(M_t^{s,tl}) \quad (10)$$

where $H_t^{s,tl} = \{h_{t,1}^{s,tl}, \dots, h_{t,t-1}^{s,tl}, h_{t,t}^{s,tl}\}$ means the mutual interaction information. $h_{t,1}^{s,tl}$ means the s slot-related representation of the 1st dialogue turn after turn interaction, when the dialogue comes to the t -th turn. The transformer encoder utilizes positional encoding to record the position information and self-attention to get interacted information in each dialogue turn.

3.4 Fusion Gate

Fusion gate is applied to merge the information as follows:

$$g_t^{s,tl} = \sigma(W_{tl} \odot [c_t^s; l_t^s]) \quad (11)$$

$$m_t^{s,tl} = (1 - g_t^{s,tl}) \otimes c_t^s + g_t^{s,tl} \otimes l_t^s \quad (12)$$

\odot and \otimes mean the matrix product and point-wise product. σ is the sigmoid function. $g_t^{s,tl}$ is the output weight of the fusion gate to keep the information from the last dialogue state. $M_t^{s,tl} = \{m_1^{s,tl}, \dots, m_{t-1}^{s,tl}, m_t^{s,tl}\}$ is the turn-level information;

$$g_t^{s,pl} = \sigma(W_{pl} \odot [m_t^{s,tl}; m_t^{s,pl}]) \quad (13)$$

$$f_t^{s,pl} = (1 - g_t^{s,pl}) \otimes m_t^{s,pl} + g_t^{s,pl} \otimes m_t^{s,tl} \quad (14)$$

$g_t^{s,pl}$ is the weight to balance the turn-level merged information $m_t^{s,tl}$ and the passage-level extracted information $m_t^{s,pl}$;

$$g_t^{s,pl'} = \sigma(W_{pl'} \odot [f_t^{s,pl}; f_t^{s,pl'}]) \quad (15)$$

$$f_t^s = (1 - g_t^{s,pl'}) \otimes f_t^{s,pl} + g_t^{s,pl'} \otimes f_t^{s,pl'} \quad (16)$$

$g_t^{s,pl'}$ is the weight to balance the merged turn/passage-level balanced information $f_t^{s,pl}$ and the enhanced result $f_t^{s,pl'}$ from equation 9. f_t^s is s slot-related core feature from context and the entire history of dialogue states.

3.5 Loss Function

Here we follow Shan et al. (2020) to calculate the probability distribution of value v_t and predict whether the slot s should be updated or kept compared to the last dialogue states. Thus our loss functions are as follows:

$$o_t^s = \text{LayerNorm}(\text{Linear}(\text{Dropout}(f_t^s))) \quad (17)$$

$$p(v_t | U_{\leq t}, R_{\leq t}, s) = \frac{\exp(-\|o_t^s - h^v\|_2)}{\sum_{v' \in \mathcal{V}_s} \exp(-\|o_t^s - h^{v'}\|_2)} \quad (18)$$

$$\mathcal{L}_{dst} = \sum_{s \in \mathcal{S}} \sum_{t=1}^T -\log(p(\hat{v}_t | U_{\leq t}, R_{\leq t}, s)) \quad (19)$$

\mathcal{L}_{dst} is the distance loss for true value \hat{v} of slot s ;

$$c_t^{s,stp} = \tanh(W_c \odot f_t^s) \quad (20)$$

$$p_t^{s,stp} = \sigma(W_p \odot [c_t^{s,stp}; c_{t-1}^{s,stp}]) \quad (21)$$

$$\mathcal{L}_{stp} = \sum_{s \in \mathcal{S}} \sum_{t=1}^T -y_t^{s,stp} \cdot \log(p_t^{s,stp}) \quad (22)$$

\mathcal{L}_{stp} is the loss function for state transition prediction, which has the value set $\{\text{keep}, \text{update}\}$. $p_t^{s,stp}$ is update probability for slot s at the t -th turn. $y_t^{s,stp}$ is the state transition label with *update* $y_t^{s,stp} = 1$ and *keep* $y_t^{s,stp} = 0$. We optimize the sum of above loss in the training process:

$$\mathcal{L}_{joint} = \mathcal{L}_{dst} + \mathcal{L}_{stp} \quad (23)$$

4 Experiments Setup

4.1 Datasets

We evaluate our model on MultiWOZ 2.0 and MultiWOZ 2.1 datasets. They are multi-domain task-oriented dialogue datasets. MultiWOZ 2.1 identified and fixed many erroneous annotations and user utterances (Zang et al., 2020).

4.2 Baseline

We compare FPDSC with the following approaches:

TRADE is composed of an utterance encoder, a slot-gate, and a generator. The approach generates value for every slot using the copy-augmented decoder (Wu et al., 2019).

CHAN employs a contextual hierarchical attention network to enhance the DST. The method applies an adaptive objective to alleviate the slot imbalance problem (Shan et al., 2020).

Model	MultiWOZ 2.0	MultiWOZ 2.1
	Joint Acc (%)	Joint Acc (%)
TRADE (Wu et al., 2019)	48.62	46.00
DST-picklist (Zhang et al., 2020a)	54.39	53.30
TripPy (Heck et al., 2020)	-	55.30
SimpleTOD (Hosseini-Asl et al., 2020)	-	56.45
CHAN (Shan et al., 2020)	52.68	58.55
CHAN* (Shan et al., 2020)	-	57.45
FPDSC (base)	51.03	54.91
FPDSC (passage-level)	52.31	55.86
FPDSC (turn-level)	55.03	57.88
FPDSC (dual-level)	53.17	59.07

Table 2: Joint accuracy on the test sets of MultiWOZ 2.0 and 2.1. CHAN* means performance without adaptive objective fine-tuning, which solves the slot-imbalance problem. CHAN means performance with the full strategy. The overall structure of FPDSC (dual-level) is illustrated in Figure 3.

DST-picklist adopts a BERT-style reading comprehension model to jointly handle both categorical and non-categorical slots, matching the value from ontologies (Zhang et al., 2020a).

TripPy applies three copy mechanisms to get value span. It regards user input, system inform memory and previous dialogue states as sources (Heck et al., 2020).

SimpleTOD is an end-to-end approach and regards sub-tasks in the task oriented dialogue task as a sequence prediction problem (Hosseini-Asl et al., 2020).

4.3 Training Details

Our code is public ², which is developed based on CHAN’s code ³. In our experiments, we use the Adam optimizer (Kingma and Ba, 2015). We use a batch size of 2 and maximal sequence length of 64 for each dialogue turn. The transformer encoder has 6 layers. The multi-head attention units have counts of 4 and hidden sizes of 784. The training process consists of two phases: 1) teacher-forcing training; 2) uniform scheduled sampling (Bengio et al., 2015). The warmup proportion is 0.1 and the peak learning rate is 1e-4. The model is saved according to the best joint accuracy on the validation data. The training process stops with no improvement in 15 continuous epochs. Our training devices are GeForce GTX 1080 Ti and Intel Core i7-6800 CPU@3.40GHZ. The training time of an epoch takes around 0.8 hour in the teacher-forcing phase and 1.6 hours in the uniform scheduled sampling phase with a GPU.

²<https://github.com/helloacl/DST-DCPDS>

³<https://github.com/smartyfh/CHAN-DST>

Deleted-Value			
Base	Turn	Passage	Dual
2.84%	22.87%	23.98%	25.22%
Related-Slot			
Base	Turn	Passage	Dual
46.63%	57.85%	62.23%	70.85%

Table 3: Success change rate of the deleted-value and related-slot experiment for FPDSC. Turn, Passage, Dual mean turn-level, passage-level and dual-level FPDSC.

5 Results and Analysis

5.1 Main Results

We use the joint accuracy to evaluate the general performance. Table 2 shows that our models get 55.03% and 59.07% joint accuracy with improvements (0.64% and 0.52%) over previous best results on MultiWOZ 2.0 and 2.1. All of our approaches get better performance on 2.1 than 2.0. This is probably because of fewer annotations error in MultiWOZ 2.1. Though table 3 shows that the passage-level variant performs better than the turn-level variant in the deleted-value and the related-slot test, passage-level variant gets worse results in the general test. The small proportion of the above problem in the MultiWOZ dataset and the strong sensitivity of the turn-level fusion gate to signal tokens in the utterance explain the phenomenon.

5.2 The Comparative Experiment for the Fusion Gate

We design a comparative network to validate the effectiveness of the turn-level fusion gate. Figure 4 shows the part structure of the comparative network (no turn-level fusion gate). The rest of the comparative network is the same as the FPDSC

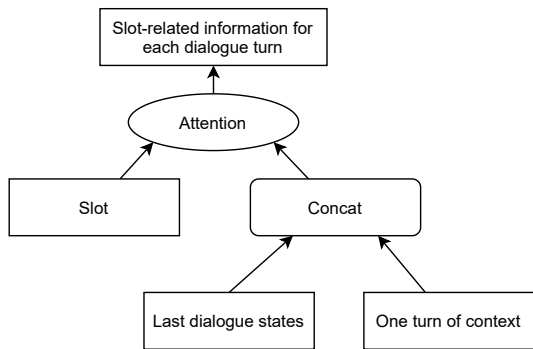


Figure 4: A part structure of the comparative network

Dataset	no-gate*	no-gate	turn-level*	turn-level
dev	46.38	52.58	56.17	61.39
test	43.03	49.24	54.08	57.88

Table 4: Joint accuracy of the comparative network (no-gate) and the FPDSC (turn-level) on the MultiWOZ 2.1 dataset. * indicates the approach is only trained with teacher-forcing, otherwise is trained by uniform scheduled sampling after teacher-forcing.

(turn-level). The table 4 shows the performance of the comparative network and the FPDSC (turn-level) on the MultiWOZ 2.1. The result validates the effectiveness of the fusion gate to merge the different information sources.

5.3 The Comparative Experiment for the Complete Dialogue States

We design another comparative network to validate the effectiveness of the complete previous dialogue states. As Figure 3 shows, $\{m_1^{s,tl}, \dots, m_{t-1}^{s,tl}, m_t^{s,tl}\}$ are fed to the transformer encoder in the FPDSC (turn-level). In the comparative network (single), $\{c_1^s, \dots, c_{t-1}^s, m_t^{s,tl}\}$ are fed to the transformer encoder. Table 5 shows the complete previous dialogue states improve the general performance of the model.

5.4 Deleted-value Tests

We select dialogues containing the deleted-value problem from test data in MultiWOZ 2.1. We re-

Dataset	single*	single	turn-level*	turn-level
dev	57.25	60.94	56.17	61.39
test	54.40	56.70	54.08	57.88

Table 5: Joint accuracy of the comparative network (single) and the FPDSC (turn-level) on the MultiWOZ 2.1 dataset. * means that the approach is only trained in the teacher-forcing training, otherwise is trained by uniform scheduled sampling training after teacher-forcing training.

U_1 : Find me a museum please restaurant-name: None
S_2 : There are 23 museums. Do you have an area as preference? U_2 : I just need the area and address for one of them. restaurant-name: None
S_3 : I have the broughton house gallery in the centre at 98 king street. U_3 : Thank you so much. I also need a place to dine in the centre that serves chinese food. restaurant-name: None
S_4 : I have 10 place in the centre. Did you have a price range you were looking at? U_4 : I would like the cheap price range. restaurant-name: None
S_5 : I recommend the rice house. Would you like me to reserve a table? U_5 : yes, please book me a table for 9 on monday at 19:30. restaurant-name: rice house
S_6 : Unfortunately, I could not book the rice house for that day and time. Is there another day or time that would work for you? U_6 : Can you try a half hour earlier or later and see if the have anything available? restaurant-name: rice house Dual-level: restaurant-name: rice house Base: restaurant-name: rice house
S_7 : No luck, would you like me to try something else? U_7 : Yes, please find another cheap restaurant for that amount of people at that time. restaurant-name: None Dual-level: restaurant-name: None Base: restaurant-name: rice house

Table 6: Dialogue id MUL2359 from MultiWOZ 2.1

gard the above dialogues as templates and augment the test data by replacing the original slot value with other slot values in the ontology. There are 800 dialogues in the augmented data. We only count the slots in dialogue turn, which occurs the deleted-value problem. As shown in Table 6, if *restaurant-name=rice house* at the 6th turn and *restaurant-name=None* at the 7th turn, we regard it as a successful tracking. We use the successful change rate to evaluate the effectiveness. Table 3 shows that the explicit introduction of the previous dialogue states in both turn-level and passage-level helps solve the problem.

5.5 Related-slot Tests

We focus on the multi-domain dialogues which contain dialogue domain of taxi for the related-slot test. We select 136 dialogue turns from the MultiWOZ 2.1 test data, which contains the template such as *book a taxi from A to B* or *commute between A and B*. We replace the explicit expression in order to focus on the actual related-slot filling situation. For example, in the dialogue from table 7, we replace the value *Ballare* to *attraction* in the user utterance

U_1 : Can you give me information on an attraction called ballare? taxi-departure: None;taxi-destination: None
S_2 : The Ballare is located in Heidelberg Gardens, Lion Yard postcode cb23na, phone number is 01223364222. The entrance fee is 5 pounds. U_2 : Thanks. I'm also looking for somewhere to stay in the north. It should be in the moderate price range and has a star of 2 as well taxi-departure: None;taxi-destination: None
S_3 : Would you want to try the lovell lodge, which is in the moderate price range and in the north. U_3 : Let's do that. Please reserve it for 6 people and 5 nights starting from thursday. taxi-departure: None;taxi-destination: None
S_4 : The booking goes through and the reference number is TY5HFLY1. U_4 : Can you help me to book a taxi from the hotel to the Ballare. I want to leave by 17:30. taxi-departure: lovell lodge taxi-destination: ballare;taxi-leave: 17:30

Table 7: Dialogue id MUL2657 from MultiWOZ 2.1

Joint Acc %	Normal Evaluation		Evaluation with Teacher Forcing	
	dev	test	dev	test
Dataset				
Base	58.01	54.91	—	—
Turn-level*	56.17	54.08	69.13	65.82
Turn-level	61.39	57.88	—	—
Passage-level*	55.21	52.40	66.84	61.92
Passage-level	61.11	55.86	—	—
Dual-level*	56.17	54.08	70.22	67.17
Dual-level	61.89	59.07	—	—

Table 8: Joint accuracy results of variants of our approach in different training phase on MultiWOZ 2.1. Normal evaluation means that the approach uses predicted dialogue states as inputs. Evaluation with teacher forcing means that it uses truth label as previous dialogue states. * means that the approach is only trained in teacher-forcing training otherwise is trained by uniform scheduled sampling training after teacher-forcing training.

at the 4th turn. We only count slots *taxi-departure* and *taxi-destination* without value of *None* in the dialogue turns, which contain the related-slot phenomenon. We divide the sum of successful tracking counts by the number of the above slots to get the success change rate. Table 3 shows the result.

5.6 Gate Visualization

Figure 5 shows the output weight of the turn/passage-level fusion gates in dialogue MUL2359 (Table 6) and MUL2657 (Table 7) from MultiWOZ 2.1. **Turn, Passage, Dual** in titles of subplots represent FPDSC with turn-level, passage-level, and dual-level. All the weights in Figure 5 mean the information keep-proportion from the last dialogue states.

When we focus on the slot **restaurant-name** in dialogue MUL2359. The output weight in the turn-level fusion gate is small at the 5th and the 7th dialogue turn in turn/dual-level approaches. Since the slot value **rice house** is first mentioned at the 5th turn and the constraint is released at the 7th turn, the change of the weight for slot **restaurant-name** is reasonable. When we focus on slots **taxi-departure**, **taxi-destination**, and **taxi-leave at** at the 4th turn of dialogue MUL2657, the respective information sources for above three slots are only previous dialogue state (**hotel-name** to **taxi-departure**), both previous dialogue state and current user utterance (**Ballare** can be found in both user utterance and previous dialogue states of **attraction-name**), only user utterance (**17:30** appears only in the user utterance at the 4th dialogue turn). As shown in Figure 5, at the 4th dialogue turn of MUL2657, **taxi-departure** has a large weight, **taxi-destination** has a middle weight, **taxi-leave at** has a small weight. This trend is as expected.

Figure 5 also shows that the turn-level fusion gate is sensitive to signal tokens in the current user expression. At the 4th dialogue turn of MUL2359, the word **cheap** triggers low output weight of the turn-level fusion gate for slots **hotel-price range** and **restaurant-price range**. It is reasonable that no domain signal is in the 4th utterance. The output of the passage-level fusion gate will keep a relatively low weight once the corresponding slot is mentioned in the dialogue except for the name-related slot.

Although the output weights of the passage-level fusion gate share similar distribution in passage/dual-level method at the 7th dialogue turn of MUL2359. FPDSC (passage-level) has a false prediction of **restaurant-name** and FPDSC (dual-level) is correct. Two fusion gates can work together to improve the performance. It explains the high performance in dual-level strategy.

5.7 Ablation Study

Table 2 shows that the passage/turn/dual-level approaches get improvements (0.95%, 2.97%, 4.16%) compared to the base approach in MultiWOZ 2.1. The results show the turn-level fusion gate is vital to our approaches. The entire history of dialogue states is helpful for DST. The uniform scheduled sampling training is crucial to improve our models' performance. In Table 8, **dev** and **test** represent validation and test data. As the table

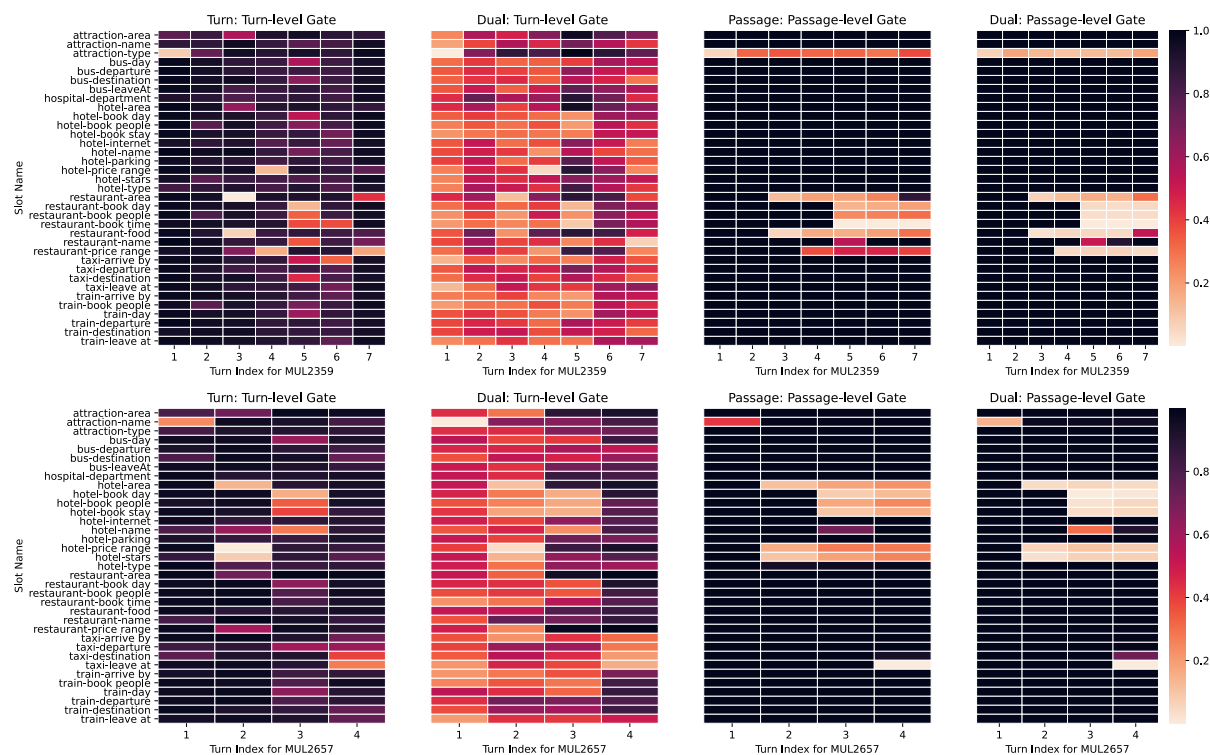


Figure 5: The output weights in the fusion gates. The weight represents the proportion of the information from the previous dialogue states. The large weight with dark color means that the prediction of the slot value pays much attention to the previous dialogue states. Turn, Passage, Dual mean FPDSC with turn-level, passage-level and dual-level.

shows, all of our approaches improve the joint accuracy around 3% after uniform scheduled sampling training. The falsely predicted dialogue states work as the data noise, which improves the model’s robustness. The base approach utilizes only the information from the context without uniform scheduled sampling training.

6 Conclusion

In this paper, we combine the entire history of the predicted dialogue state and the contextual representation of dialogue for DST. We use a hierarchical fusion network to merge the turn/passage-level information. Both levels of information is useful to solve the deleted-value and related-slot problem. Besides, our models reach state-of-the-art performance on MultiWOZ 2.0 and MultiWOZ 2.1.

The turn-level fusion gate is sensitive to signal tokens from the current turn utterance. The passage-level fusion gate is relatively stable. Uniform scheduled sampling training is crucial to improve the performance. The entire history of dialogue states helps at extracting information in each dialogue utterance. Although error exists in the predicted dialogue states, the errors work as the data noise in

the training to enhance the model’s robustness.

Although our approach is based on predefined ontology, the strategy for information extraction is universal. Besides, the core feature f_t^s can be introduced to a decoder to generate the slot state, which suits the open-domain DST.

Acknowledgement

We thank the anonymous reviewers for their helpful comments. This work was supported by the NSFC projects (No. 62072399, No. 61402403), Hithink RoyalFlush Information Network Co., Ltd, Hithink RoyalFlush AI Research Institute, Chinese Knowledge Center for Engineering Sciences and Technology, MoE Engineering Research Center of Digital Library, and the Fundamental Research Funds for the Central Universities.

References

- Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. [Scheduled sampling for sequence prediction with recurrent neural networks](#). In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Informa-*

- tion Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada, pages 1171–1179.
- 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.
- 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.
- Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishhauser, Hsien-Chin Lin, Marco Moresi, and Milica Gasic. 2020. [TripPy: A triple copy strategy for value independent neural dialog state tracking](#). In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 35–44, 1st virtual meeting. Association for Computational Linguistics.
- Matthew Henderson, Blaise Thomson, and Steve Young. 2014. [Word-based dialog state tracking with recurrent neural networks](#). In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 292–299, Philadelphia, PA, U.S.A. Association for Computational Linguistics.
- Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. [A simple language model for task-oriented dialogue](#). In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.
- 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.
- Diederik P. Kingma and Jimmy Ba. 2015. [Adam: A method for stochastic optimization](#). In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Shikib Mehri, Mihail Eric, and Dilek Hakkani-Tur. 2020. Dialogue: A natural language understanding benchmark for task-oriented dialogue. *arXiv preprint arXiv:2009.13570*.
- Yawen Ouyang, Moxin Chen, Xinyu Dai, Yinggong Zhao, Shujian Huang, and Jiajun Chen. 2020. [Dialogue state tracking with explicit slot connection modeling](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 34–40, Online. Association for Computational Linguistics.
- Yong Shan, Zekang Li, Jinchao Zhang, Fandong Meng, Yang Feng, Cheng Niu, and Jie Zhou. 2020. [A contextual hierarchical attention network with adaptive objective for dialogue state tracking](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6322–6333, Online. Association for Computational Linguistics.
- Blaise Thomson and Steve Young. 2010. Bayesian update of dialogue state: A pomdp framework for spoken dialogue systems. *Computer Speech & Language*, 24(4):562–588.
- 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.
- Zhuoran Wang and Oliver Lemon. 2013. [A simple and generic belief tracking mechanism for the dialog state tracking challenge: On the believability of observed information](#). In *Proceedings of the SIGDIAL 2013 Conference*, pages 423–432, Metz, France. Association for Computational Linguistics.
- Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. [A network-based end-to-end trainable task-oriented dialogue system](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.
- Jason D. Williams. 2014. [Web-style ranking and SLU combination for dialog state tracking](#). In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 282–291, Philadelphia, PA, U.S.A. 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, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019. [Transferable multi-domain state generator for task-oriented dialogue systems](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 808–819, Florence, Italy. Association for Computational Linguistics.
- Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. [MultiWOZ 2.2 : A dialogue dataset with](#)

- additional annotation corrections and state tracking baselines. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 109–117, Online. Association for Computational Linguistics.
- Jianguo Zhang, Kazuma Hashimoto, Chien-Sheng Wu, Yao Wang, Philip Yu, Richard Socher, and Caiming Xiong. 2020a. [Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking](#). In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 154–167, Barcelona, Spain (Online). Association for Computational Linguistics.
- Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020b. [DIALOGPT : Large-scale generative pre-training for conversational response generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.
- Su Zhu, Jieyu Li, Lu Chen, and Kai Yu. 2020. [Efficient context and schema fusion networks for multi-domain dialogue state tracking](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 766–781, Online. Association for Computational Linguistics.
- Lukas Zilka and Filip Jurcicek. 2015. Incremental lstm-based dialog state tracker. In *2015 Ieee Workshop on Automatic Speech Recognition and Understanding (Asru)*, pages 757–762. IEEE.