Generation and Extraction Combined Dialogue State Tracking with Hierarchical Ontology Integration

Xinmeng Li and **Qian Li** and **Wansen Wu** and **Quanjun Yin**
College of Systems Engineering, National University of Defense Technology
xml.nudt@gmail.com, liqian9510@outlook.com, {wuwansen14, yinquanjun}@nudt.edu.cn

Abstract

Recently, the focus of dialogue state tracking has expanded from single domain to multiple domains. The task is characterized by the shared slots between domains. As the scenario gets more complex, the out-of-vocabulary problem also becomes more severe. Current models are not satisfactory for addressing the challenges of ontology integration between domains and out-of-vocabulary problems. To address the problem, we explore the hierarchical semantics of the ontology and enhance the interrelation between slots with masked hierarchical attention. In state value decoding stage, we address the out-of-vocabulary problem by combining generation method and extraction method together. We evaluate the performance of our model on two representative datasets, MultiWOZ in English and CrossWOZ in Chinese. The results show that our model yields a significant performance gain over current state-of-the-art state tracking model and it is more robust to out-of-vocabulary problem compared with other methods.

1 Introduction

Dialogue state tracking (DST) is in charge of updating the belief state in task-oriented dialogue system (Gao et al., 2019a). Traditional discriminative DST models assume that the task ontology is well defined in advance, that is to say, all states and their values are known to the model. They usually rely on hand-crafted features or task-specific lexicon (Henderson et al., 2014; Mrkšić and Vulić, 2018). An inconvenience is that they are time-consuming and hard to expand to new tasks. To overcome it, the open vocabulary-based models are proposed to decode the state value according to the dialogue context (Xu and Hu, 2018; Jin et al., 2018; Lei et al., 2018; Goel et al., 2019).

In recent years, the research frontier for task-oriented dialogue systems has expanded from single domain to multiple domains (Budzianowski et al., 2018; Zhu et al., 2020; Cheng et al., 2020). There come new challenges demanding prompt solution. Firstly, current model does not sufficiently consider the interrelation between slots in multi-domain scenario. For example, user asks "I also want to find an attraction near the restaurant", which implies that the hotel need to have the same *area* with the restaurant. The implicit relation between the *area* slot of *hotel* and *restaurant* is the key to exactly track user's intent (Wu et al., 2019). Prior work simply used the summed (Wu et al., 2019) or concatenated (Zhang et al., 2019; Kim et al., 2020) embedding of the domain and slot as the states representation for the decoder. Secondly, out-of-vocabulary (OOV) problem gets more severe since the user asking question with wider entities and more diverse words.

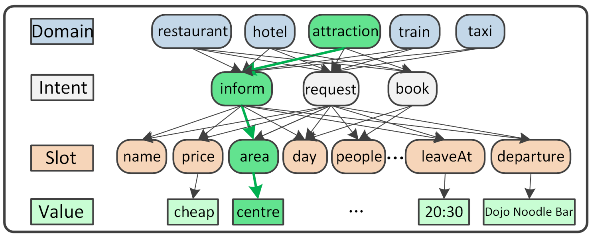


Figure 1: The hierarchically represented ontology for MultiWOZ dataset. Each of the states can be represented by an arrow line connecting through *Domain*, *Intent* and *Slot*.

In this paper, we propose the generation and extraction combined method with hierarchical ontology integration, named *GeeX*, for dialogue state tracking. First, we explore the hierarchical semantics of the ontology to enhance the representation of slots in multiple domains. Inspired by Chen et al. (2019), we adopt the directed acyclic graph to represent the ontology and enhance the slots interaction between domains with masked hierarchical attention. We use the ontology of MultiWOZ (Budzianowski et al., 2018) to illustrate this mechanism. As shown in figure 1, the ontology has four

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 2241–2249
November 7–11, 2021.

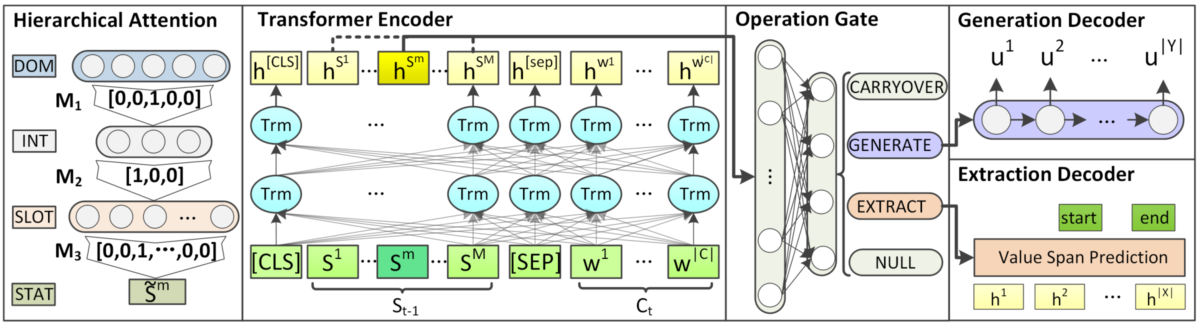


Figure 2: The architecture of GeeX. The model includes masked hierarchical attention, Transformer-based encoder, fully-connection operation gate and generation and extraction combined state value decoder.

hierarchies, i.e., *Domain*, *Intent*, *Slot* and *Value*. The states can be expressed as the combination of *Domain*, *Intent* and *Slot* and the goal of DST is to decode the *Value* for the state mentioned in the dialogue context. The hierarchically represented ontology is efficient and effective in two aspects. First, it enhances the interrelation between slots in multiple domains. Second, the compact structure is efficient for state representation which is appropriate to domain expansion since new domain often shares slots with old one (Rastogi et al., 2020). To address the OOV problem, we leverage generation and extraction by combining the two methods together. We first predict the state operation policy to select the suitable decoding strategy. Then, we enter into the corresponding decoder for value decoding according to the predicted policy.

The contributions of this paper are summarized as follows: (i) We adopt the masked hierarchical attention to represent the ontology to enhance the slots interrelation between domains. (ii) We combine generation and extraction to handle OOV problem in dialogue state tracking. (iii) Experiment results demonstrate that GeeX outperforms state-of-the-art baseline on two representative datasets. Furthermore, GeeX also shows robustness in OOV testing.

2 Architecture

We use a four-stage model for state tracking, Figure 2 illustrates the architecture.

2.1 Masked Hierarchical Attention

We use a three-layer masked hierarchical attention to explicitly integrate the state information. Assuming there are M states in total[1]. For the m-th state, we use a state-specific mask $M_l^m \in R^{|M_l|}$ to activate certain gate and only pass through their information to the next level to disentangle the layer-wise information[2]. The state is computed by,

$$\begin{aligned} \tilde{S}_l^m &= \sum\nolimits_{|M_l|} M_l \operatorname{Att}(\tilde{S}_{l-1}^m, \tilde{O}_l, \tilde{O}_l) \in R^{d_h} \\ \text{where}, \tilde{O}_l &= \operatorname{Att}(O_l, O_l, O_l) \in R^{|M_l| \times d_h} \end{aligned} \tag{1}$$

where Att is the standard scaled dot-product attention (Vaswani et al., 2017), l is the layer number, d_h is the hidden dimension, $|\cdot|$ denotes the length number. O_l represents the layer of *Domain*, *Intent* and *Slot* when $l = 1, 2, 3$, respectively. The dialogue state is the concatenation of all individual states, i.e., $S = S^1 \oplus \cdots \oplus S^M$, where $S^m = \tilde{S}^m \oplus V^m$, V^m is the value of $\tilde{S}^m$ and $\oplus$ denotes the concatenation operation. Note that they are shareable among layers, so the hierarchical attention helps to implicitly model the interrelation between states.

2.2 Transformer Encoder

We represent the dialogue context as the concatenation of last turn system response D and current turn user utterance U[3]. At t-th turn , the dialogue context is denoted as $C_t = D_{t-1} \oplus U_t$. We use Transformer (Vaswani et al., 2017) to fuse the state information into dialogue context. We concatenate last turn state S_{t-1} and current dialogue context C_t as the input, i.e., $X_t = [CLS] \oplus S_{t-1} \oplus [SEP] \oplus C_t$, where $[CLS]$ and $[SEP]$ are the special token as in (Devlin et al., 2019). In output layer, we get the hidden representation for each of the input tokens.

[1]The full states in MultiWOZ and CrossWOZ are listed in Appendix A.

[2]The mask is a one-hot representation for *Domain*, *Intent* and *Slot*, respectively. For example, 'attraction-inform-area' is denoted as [0,0,1,0,0], [1,0,0], [0,0,1,...,0,0] in each layer, as shown in Figure 2.

[3]To efficiently track current turn dialogue state, we adopt the selectively overwriting framework (Kim et al., 2020) to take advantage of last turn states.

Particularly, h^{S^m} corresponds to $\tilde{S}^m$ representing the information of the m-th state.

2.3 Operation Gate

We predict the decoding operation $o_k \in O = \{CARRYOVER, GENERATE, EXTRACT, NULL\}$ with a four-channel classifier, where, *CARRYOVER* denotes to keep the state value the same as last turn, *GENERATE* represents to decode the value by generation decoding, *EXTRACT* represents to decode the value by extraction decoding, and *NULL* means the state is not mentioned in the context and its value is empty. For each of the states, we compute the decoding operation probability $\mathcal{P}_{op}^m \in R^{|O|}$ by,

$$\mathcal{P}_{op}^m = \text{Softmax}(W_{op} h^{S^m}) \tag{2}$$

where, $W_{op} \in R^{|O| \times d_h}$ is a learnable parameter.

2.4 State Value Decoder

We build two parallel decoders for state value prediction and selectively decode the value for states whose operation policy is *GENERATE* and *EXTRACT*. For each of the states, when its policy is *GENERATE*, we execute the generation decoding mode, and when its policy is *EXTRACT*, we enter into extraction decoding mode.

Generation Decoding We use Gated Recurrent Units (GRU) (Cho et al., 2014) as the basic decoder and employ copy mechanism to calculate a probability over the dialogue context to encourage reusing words in the context. We use the state representation h^{S^m} (whose operation policy is *GENERATE*) to initialize the decoder hidden. The final probability of decoding a certain word, e.g., the τ-th token $u(\tau)$, is calculated by

$$\mathcal{P}_{gen}^m(\tau) = \mathcal{P}_{voc}(\tau) + \mathcal{P}_{copy}(\tau) \tag{3}$$

where $\mathcal{P}_{voc}(\tau)$ is the probability computed from the decoder hidden over whole vocabulary, and $\mathcal{P}_{copy}(\tau)$ indicates the probability of copying words from the context.

Extraction Decoding We treat the state value prediction as the extractive reading comprehension problem (Gao et al., 2019b, 2020). Specifically, we use the state representation h^{S^m} (whose policy is *EXTRACT*) as the query, the dialogue context H as background and the state value as the answer. The extraction can be formalized as

$$\mathcal{P}_s^m, \mathcal{P}_e^m = \text{EXT}(h^{S^m}, H) \tag{4}$$

where, $\mathcal{P}_s^m$ and $\mathcal{P}_e^m$ are the start index probability and the end index probability over the dialogue context, respectively. In implementation, we use the extraction method EXT from (Hu et al., 2018).

2.5 Learning

We use cross-entropy to compute the operation policy loss and state value decoding loss:

$$\begin{aligned}\mathcal{L}_{op} &= -\sum_{m=1}^{M} y_{op}^m \log \mathcal{P}_{op}^m \\ \mathcal{L}_{gen} &= -\sum_{GEN}\sum_{\tau} y_{gen}^m(\tau) \log \mathcal{P}_{gen}^m(\tau) \\ \mathcal{L}_{ext} &= -\sum_{EXT} y_s^m \log \mathcal{P}_s^m + y_e^m \log \mathcal{P}_e^m\end{aligned} \tag{5}$$

where y_*^* is the standard label for P_*^*. We adopt multi-task learning to train the model. The optimization objective is a combination of the three loss function,

$$\mathcal{L} = \mathcal{L}_{op} + \mathcal{L}_{gen} + \mathcal{L}_{ext} \tag{6}$$

3 Experiment

3.1 Settings

Dataset We conduct experiments on MultiWOZ2.0 (Budzianowski et al., 2018) MultiWOZ2.1[4] (Eric et al., 2019) and CrossWOZ (Zhu et al., 2020). MultiWOZ is the most representative multi-domain task-oriented dialogue datasets and CrossWOZ is the latest multi-domain task-oriented dialogue datasets in Chinese.[5]

Evaluation Metric We adopt Joint State Accuracy to evaluate the model performance, which checks whether all state values exactly match the ground truth values at each dialogue turn.

Training Detail We optimize GeeX with Adam (Kingma and Ba, 2015). The hidden size is set to 300. The learning rate is initialized to 10^{-3} and annealed in the range of [10^{-3}, 10^{-5}] with a decay rate of 0.5.

Benchmark We compare GeeX with both discriminative methods i.e., SUMBT (Lee et al., 2019) and open vocabulary-based method, i.e., DSTReader (Gao et al., 2019b), TRADE (Wu et al., 2019), SOM (Kim et al., 2020), TripPy (Heck et al., 2020). To further verify the effectiveness of hierarchical ontology integration and parallel decoding strategy, we also design three ablation

[4]MultiWOZ2.1 shares the same dialogues with MultiWOZ 2.0 but it fixed previous annotation errors.

[5]Note that there is no gold operation label in datasets, so we automatically annotate it, as described in Appendix B.

models here, (i) –MHA. It replaces the masked hierarchical attention with flat representation as in SOM. (ii) –EXT. It removes the extraction decoding branch. All the state values are decoded with generation method. (iii) –GEN. It removes the generation decoding branch. All the state values are decoded with extraction method.

3.2 Results and Analysis

3.2.1 Multi-domain Testing

As illustrated in Table 1, GeeX outperforms all the baselines on MultiWOZ2.0, MultiWOZ2.1 and CrossWOZ.

The performance difference between vanilla extractive model (i.e., DSTReader, TripPy) and GeeX mainly comes from the limitation that its decoding vocabulary is limited to the words that occurred in the dialogue history. For examples, a user may find a cheap restaurant while described it as economical, the extractive model would lose efficacy to predict the right answer span.

GeeX also achieves higher score than the generation decoding models (i.e., TRADE, SOM). After further observations, we find that most of the token can be directly extracted from context (82.0% in MultiWOZ2.0, 84.2% in MultiWOZ2.1 and 83.7% in CrossWOZ). The extractive decoding models is more robust to decode longer sequence. However, the generation decoding method helps to generate values not appearing in the context, so it is a perfect complement to extractive method.

Another performance gain comes from operation prediction. As stated in (Kim et al., 2020), a relatively larger amount of error originates from operation gate. SOM uses *CARRYOVER* for states keeping unchanged while neglecting the difference between "none" and succeeding. GeeX use *CARRYOVER* for state value succeeding from last turn and *NULL* for empty value, which help to explicitly take advantage of last turn belief states.

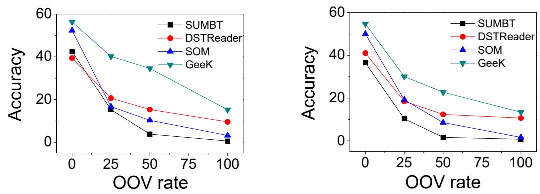
3.2.2 Ablation Study

Ablation results are reported in bottom half of Table 1, the degradation of –MHA, –EXT and –GEN validates the necessity of hierarchical ontology integration and parallel decoding approach. –EXT outperforms SOM in generation decoding method and –GEN outperforms DSTReader in extraction decoding method, demonstrating that hierarchical ontology integration is effective to promote the slots interaction and lead to the performance boost. Compared with vanilla extraction and generation decoding models, the improvement on –MHA further clarifies that the two parallel decoding approaches are complementary to each other.

Model	MultiWOZ2.0	MultiWOZ2.1	CrossWOZ
SUMBT*	42.40	42.40	36.56
DSTReader$^{\diamond}$	39.41	36.40	41.04
TRADE†	48.62	45.60	36.08
SOM†	52.32	52.57	50.06
TripPy$^{\diamond}$	\	55.29	\
GeeX$^{\diamond\dagger}$	**56.35**	**56.42**	**54.70**
–MHA$^{\diamond\dagger}$	53.41	52.73	51.98
–EXT†	54.39	50.95	51.23
–GEN$^{\diamond}$	47.86	51.64	50.22

Table 1: Model test set performance (%). $\star$ denotes the discriminative model. $\diamond$ and $\dagger$ denote open vocabulary-based model with extraction and generation decoder, respectively. The best result is highlighted in bold.

3.2.3 OOV Testing



(a) MultiWOZ2.0 (b) CrossWOZ

Figure 3: Result on OOV testing (%). We randomly mask the words in value with the probability of 0%, 25%, 50%, 100%, respectively.

We simulate OOV instances by randomly masking the value token in dialogue context. For example, we change 'I would like *modern European* food' into 'I would like *[UNK] European* food'. Here, we take the three representative models, i.e., SUMBT, DSTReader and SOM, for comparison.

As shown in Figure 3, compared with SUMBT, DSTReader and SOM, GeeX still performs well in all OOV rates. This is actually because the extraction decoder plays a crucial role for predicting OOV tokens, which is also reflected in the smaller performance drop of DSTReader. In addition, the performance of SOM decreases more sharply as more instances set to be OOV, demonstrating that the copy-augmented model is inflexible to address multiple sequential unknown words. The worst performance of SUMBT demonstrates that the discriminative model is ill-equipped to recognize unknown tokens.

4 Conclusion

In the paper, we explore the hierarchical structure of ontology and combine generation and extraction together for state value decoding. With the domain expanding, supervised learning is not satisfactory for rapidly increasing requirements. In future work, few-shot learning and knowledge fusion can be applied to further improve domain transferring performance.

Acknowledgment

This work was partly supported by the National Natural Science Foundation of China under grant No.62103420. We thank all the reviewers for their helpful comments to improve the paper.

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.

Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. Semantically conditioned dialog response generation via hierarchical disentangled self-attention. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3696–3709, Florence, Italy. Association for Computational Linguistics.

Jianpeng Cheng, Devang Agrawal, Héctor Martínez Alonso, Shruti Bhargava, Joris Driesen, Federico Flego, Dain Kaplan, Dimitri Kartsaklis, Lin Li, Dhivya Piraviperumal, Jason D. Williams, Hong Yu, Diarmuid Ó Séaghdha, and Anders Johannsen. 2020. Conversational semantic parsing for dialog state tracking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8107–8117, Online. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1724–1734. ACL.

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

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. *CoRR*, abs/1907.01669.

Jianfeng Gao, Michel Galley, and Lihong Li. 2019a. Neural approaches to conversational ai. *Foundations and Trendső in Information Retrieval*, 13(2-3):127–298.

Shuyang Gao, Sanchit Agarwal, Tagyoung Chung, Di Jin, and Dilek Hakkani-Tür. 2020. From machine reading comprehension to dialogue state tracking: Bridging the gap. *CoRR*, abs/2004.05827.

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

Rahul Goel, Shachi Paul, and Dilek Hakkani-Tür. 2019. Hyst: A hybrid approach for flexible and accurate dialogue state tracking. In *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pages 1458–1462. ISCA.

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

Matthew Henderson, 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.

Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. 2018. Reinforced mnemonic reader for machine reading comprehension. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI*

2018, July 13-19, 2018, Stockholm, Sweden, pages 4099–4106. ijcai.org.

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*, 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.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *The 3rd International Conference on Learning Representations*.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019. SUMBT: Slot-utterance matching for universal and scalable belief tracking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5478–5483, Florence, Italy. Association for Computational Linguistics.

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

Nikola Mrkšić and Ivan Vulić. 2018. Fully statistical neural belief tracking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 108–113, Melbourne, Australia. Association for Computational Linguistics.

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

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, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

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.

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.

Jianguo Zhang, Kazuma Hashimoto, Chien-Sheng Wu, Yao Wan, Philip S. Yu, Richard Socher, and Caiming Xiong. 2019. Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking. *CoRR*, abs/1910.03544.

Qi Zhu, Kaili Huang, Zheng Zhang, Xiaoyan Zhu, and Minlie Huang. 2020. Crosswoz: A large-scale chinese cross-domain task-oriented dialogue dataset. *Trans. Assoc. Comput. Linguistics*, 8:281–295.

Appendix

A. The states of MultiWOZ and CrossWOZ

The full belief states in MultiWOZ[6] and CrossWOZ are list in Table 2.

B. The operation label annotation

There is no gold operation label in datasets, so we automatically annotate it according to the procedure below:

(i) If the state is empty, we label it with *NULL*.

(ii) If the state has value, and the value keeps unchanged compared with last turn, we label it with *CARRYOVER*.

(iii) If the state has different value compared with last turn, and the value exists in dialogue context, we label it with *EXTRACT*; otherwise, we label it with *GENERATE*.

The detailed examples are shown in table 3.

[6]MultiWOZ2.0 and MultiWOZ2.1 share the same belief states.

<table>
<tr><th></th><th>MultiWOZ</th><th>CrossWOZ</th></tr>
<tr><td>Domain</td><td>restaurant, hotel, attraction, taxi, train</td><td>景点(attraction), 餐馆(restaurant), 酒店(hotel), 地铁(metro), 出租车(taxi)</td></tr>
<tr><td>Intent</td><td>inform, book</td><td>-</td></tr>
<tr><td>Slot</td><td>area, name, type, day, people, stay, internet, parking, pricerange, stars, time, food, arriveby, departure, destination, leaveat, arriveby</td><td>出发地(from), 目的地(to), 车型(car type), 车牌(plate number), 出发地附近地铁站(from station), 目的地附近地铁站(to station), 名称(name), 周边景点(nearby attract.), 周边酒店(nearby hotels), 周边餐馆(nearby rest.), 地址(address), 游玩时间(duration), 源领域(source domain), 电话(phone), 评分(rating), 门票(ticket), 价格(cost), 酒店类型(type), 酒店设施(facility), 人均消费(consumption per person), 推荐菜(recommendation dishes), 营业时间(openning hours)</td></tr>
<tr><td>State</td><td>attraction-inform-area,
attraction-inform-name,
attraction-inform-type,
hotel-inform-area,
hotel-book-day,
hotel-book-people,
hotel-book-stay,
hotel-inform-internet,
hotel-inform-name,
hotel-inform-parking,
hotel-inform-pricerange,
hotel-inform-stars,
hotel-inform-type,
restaurant-inform-area,
restaurant-book-day,
restaurant-book-people,
restaurant-book-time,
restaurant-inform-food,
restaurant-inform-name,
restaurant-inform-pricerange,
taxi-inform-arriveby,
taxi-inform-departure,
taxi-inform-destination,
taxi-inform-leaveat,
train-inform-arriveby,
train-book-people,
train-inform-day,
train-inform-departure,
train-inform-destination,
train-inform-leaveat</td><td>出租-出发地,
出租-目的地,
出租-车型,
出租-车牌,
地铁-出发地,
地铁-出发地附近地铁站,
地铁-目的地,
地铁-目的地附近地铁站,
景点-名称,
景点-周边景点,
景点-周边酒店,
景点-周边餐馆,
景点-地址,
景点-游玩时间,
景点-源领域,
景点-电话,
景点-评分,
景点-门票,
酒店-价格,
酒店-名称,
酒店-周边景点,
酒店-周边酒店,
酒店-周边餐馆,
酒店-地址,
酒店-源领域,
酒店-电话,
酒店-评分,
酒店-酒店类型,
酒店-酒店设施,
餐馆-人均消费,
餐馆-名称,
餐馆-周边景点,
餐馆-周边酒店,
餐馆-周边餐馆,
餐馆-地址,
餐馆-推荐菜,
餐馆-源领域,
餐馆-电话,
餐馆-营业时间,
餐馆-评分</td></tr>
</table>

Table 2: The full states in MultiWOZ and CrossWOZ. Following (Wu et al., 2019), only five domains(restaurant, hotel, attraction, taxi, train) in MultiWOZ are used in our experiment.

	D_{t-1}	U_t	S_t
Turn 1	-	Hello I am looking for a place to dine in the centre of town that needs to be cheaply priced.	restaurant-inform-pricerange-cheap [*GENERATE*] restaurant-inform-area-centre [*EXTRACT*] Others [*NULL*]
Turn 2	I have 15 different restaurants available. What type of food would you like them to serve?	I would like it to have scandinavian food.	restaurant-inform-pricerange-cheap [*CARRYOVER*] restaurant-inform-area-centre [*CARRYOVER*] restaurant-inform-food-scandinavian [*EXTRACT*] Others [*NULL*]
Turn 3	Unfortunately none of them serve scandanavian food. I don't believe any restaurants in town do. Is there another cuisine you might like instead?	Yes, do any serve Modern European food instead?	restaurant-inform-pricerange-cheap [*CARRYOVER*] restaurant-inform-area-centre [*CARRYOVER*] restaurant-inform-food-Modern European [*EXTRACT*] Others [*NULL*]
Turn 4	The River Bar steakhouse and grill does. Can I make you a reservation?	No thank you. But can I get the address, phone number and postcode please?	restaurant-inform-pricerange-cheap [*CARRYOVER*] restaurant-inform-area-centre [*CARRYOVER*] restaurant-inform-food-Modern European [*CARRYOVER*] Others [*NULL*]
Turn 5	They are located at Quayside Off Bridge Street, phone number is 01223307030, and their postcode is cb58aq.	I also need a train that depart bishops stortford and goes to cambridge.	restaurant-inform-pricerange-cheap [*CARRYOVER*] restaurant-inform-area-centre [*CARRYOVER*] restaurant-inform-food-Modern European [*CARRYOVER*] train-inform-departure-bishops stortford [*EXTRACT*] train-inform-destination-cambridge [*EXTRACT*] Others [*NULL*]
Turn 6	Can you please tell me what day you would like your train and what time?	I am leaving on Monday after 13:15 in the afternoon. Please give me the reference number when you have it. Thanks!	restaurant-inform-pricerange-cheap [*CARRYOVER*] restaurant-inform-area-centre [*CARRYOVER*] restaurant-inform-food-Modern European [*CARRYOVER*] train-inform-departure-bishops stortford [*CARRYOVER*] train-inform-destination-cambridge [*CARRYOVER*] train-inform-leaveAt-13:15 [*EXTRACT*] train-inform-day-Monday [*EXTRACT*] Others [*NULL*]
Turn 7	Okay I have booked you a train ticket from Bishops stortford to Cambridge for 1 person leaving at 13:29. Your reference number is 962Q6MQG.	I apologize, I forgot to mention that I'll be needing 6 tickets, not just 1.	restaurant-inform-pricerange-cheap [*CARRYOVER*] restaurant-inform-area-centre [*CARRYOVER*] restaurant-inform-food-Modern European [*CARRYOVER*] train-inform-departure-bishops stortford [*CARRYOVER*] train-inform-destination-cambridge [*CARRYOVER*] train-inform-leaveAt-13:15 [*CARRYOVER*] train-inform-day-Monday [*CARRYOVER*] train-book-people-6 [*EXTRACT*] Others [*NULL*]
Turn 8	Okay! I've changed your reservation for 6 tickets rather than one. The reference number is now QHYYACDV.	Thanks, I can't wait! And thanks for your help today.	restaurant-inform-pricerange-cheap [*CARRYOVER*] restaurant-inform-area-centre [*CARRYOVER*] restaurant-inform-food-Modern European [*CARRYOVER*] train-inform-departure-bishops stortford [*CARRYOVER*] train-inform-destination-cambridge [*CARRYOVER*] train-inform-leaveAt-13:15 [*CARRYOVER*] train-inform-day-Monday [*CARRYOVER*] train-book-people-6 [*CARRYOVER*] Others [*NULL*]

Table 3: An example to illustrate the operation gate of GeeX. [$\cdot$] denotes the operation policy. Others denote the states which aren't mentioned in the context.