

Knowledge-Retrieval Task-Oriented Dialog Systems with Semi-Supervision

Yucheng Cai^{1,3}, Hong Liu^{1,3}, Zhijian Ou^{*,1,3}, Yi Huang^{2,3}, Junlan Feng^{2,3}

¹Speech Processing and Machine Intelligence (SPMI) Lab, Tsinghua University, Beijing, China

²China Mobile Research Institute, Beijing, China

³Tsinghua University-China Mobile Communications Group Co., Ltd. Joint Institute, Beijing, China

{cyc22, liuhong21}@mails.tsinghua.edu.cn, ozj@tsinghua.edu.cn,
{huangyi, fengjunlan}@chinamobile.com

Abstract

Most existing task-oriented dialog (TOD) systems track dialog states in terms of slots and values and use them to query a database to get relevant knowledge to generate responses. In real-life applications, user utterances are noisier, and thus it is more difficult to accurately track dialog states and correctly secure relevant knowledge. Recently, a progress in question answering and document-grounded dialog systems is retrieval-augmented methods with a knowledge retriever. Inspired by such progress, we propose a retrieval-based method to enhance knowledge selection in TOD systems, which significantly outperforms the traditional database query method for real-life dialogs. Further, we develop latent variable model based semi-supervised learning, which can work with the knowledge retriever to leverage both labeled and unlabeled dialog data. Joint Stochastic Approximation (JSA) algorithm is employed for semi-supervised model training, and the whole system is referred to as JSA-KRTOD. Experiments are conducted on a real-life dataset from China Mobile Custom-Service, called MobileCS, and show that JSA-KRTOD achieves superior performances in both labeled-only and semi-supervised settings.

Index Terms: task-oriented dialog system, knowledge retrieval, semi-supervised learning, joint stochastic approximation

1. Introduction

Task-oriented dialog (TOD) systems are designed to help users to achieve their goals through multiple turns of natural language interaction [1, 2]. The system needs to perform dialog state tracking (DST), query a task-related database (DB), decide actions and generate responses iteratively across turns. The information flow in a task-oriented dialog is illustrated in Figure 1.

Recent studies recast such information flow in TOD systems as conditional generation of tokens based on pretrained language models (PLMs) such as GPT2 [3] and T5 [4]. Fine-tuning PLMs over annotated dialog datasets via supervised learning [5, 6, 7, 8] has shown promising results on Wizard-of-Oz TOD datasets such as MultiWOZ [9]. However, the Wizard-of-Oz dialog data are in fact simulated data. In real-life applications such as custom services, user utterances are more casual and may contain noises from automatic speech recognition. It is more difficult for the dialog system to perform well on DST, and the rule-based database query is not robust to erroneous dialog states. This brings performance degradation in real-life applications.

To address the problems mentioned above, we propose to introduce a knowledge retriever into TOD systems, and the new system is called knowledge-retrieval TOD system (KRTOD). It has been shown in recent studies [10, 11, 12] that knowledge

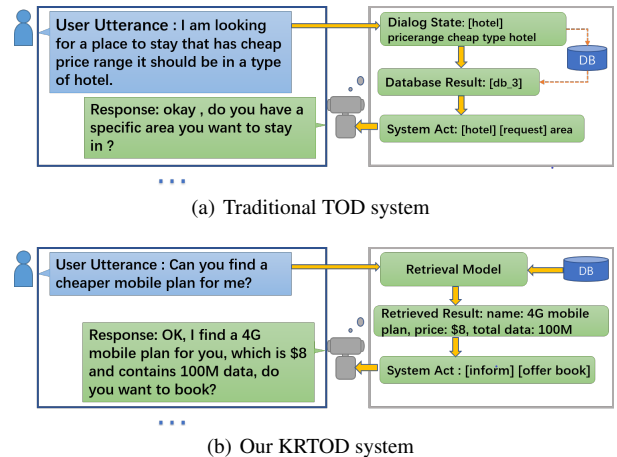


Figure 1: Comparison between our KRTOD system and the traditional TOD system. KRTOD uses a neural network based retrieval model instead of the traditional rule-based database query, and can yield more informative and successful responses.

retrievers, such as BM25 or dense passage retriever (DPR), can retrieve appropriate external knowledge based on a conversational context in question answering and knowledge grounded dialog. By introducing the retrievers, both the difficulty in correctly tracking the dialog state and the inflexibility of rule-based database query can be alleviated in KRTOD. First, through using the retriever to secure knowledge from the database, we can avoid tracking of dialog states, which originally are needed for database query. Annotations for DST may also be omitted, simplifying the procedure of building TOD systems. Second, using a retriever instead of rule-based database query makes the system more flexible in knowledge query. More complex situations, such as that illustrated in Figure 1 can be handled through a retriever.

Further, note that obtaining intermediate labels of the ground-truth knowledge is usually expensive and difficult for knowledge-grounded systems. So for KRTOD systems, we further develop latent variable model based semi-supervised learning, which can work with the knowledge retriever to leverage both labeled and unlabeled dialog data. The Joint Stochastic Approximation (JSA) algorithm [13, 14] is employed for semi-supervised model training. The whole system is called JSA-KRTOD. Intuitively, when the knowledge source is not available (i.e., over unlabeled data), the inference model in JSA learning help to infer the required knowledge from the system response.

Experiments are conducted on a real-life human-human dialog dataset, called MobileCS (Mobile Customer-Service), released from the EMNLP 2022 SereTOD Challenge [15]. It consists of real-world dialog transcripts between real users and

* Corresponding author: Zhijian Ou. The code is released at <https://github.com/thu-spmi/JSA-KRTOD>

customer-service staffs from China Mobile, which are more noisy and casual than prior Wizard-of-Oz data.

In summary, main contributions of this work are three folds:

- We propose to use a knowledge retriever to retrieve knowledge from the database instead of the traditional database query method in TOD system. The proposed KRTOD system is more suitable for real-life applications with noisier user utterances.
- Further, we propose to use the JSA algorithm to perform semi-supervised learning for KRTOD systems. The resulting JSA-KRTOD system can work with the knowledge retriever to effectively leverage both labeled and unlabeled dialog data.
- Extensive experiments are conducted on a real-life dataset, MobileCS, and show that JSA-KRTOD achieves superior performances in both labeled-only and semi-supervised settings.

2. Related Work

2.1. TOD Systems

Recent studies recast TOD systems as an end-to-end conditional generation problem based on pretrained language models (PLMs) such as GPT2 [3] and T5 [4]. Those systems concatenate user utterances, dialog states, DB results, system acts and responses and train the model to generate them autoregressively [5, 6, 7, 8], which achieves good performance on TOD datasets such as MultiWOZ [9]. Dialog state tracking [16, 17] is crucial for the whole system, and the dialog state is used to query the database to get necessary information for appropriate response generation. However, previous studies have pointed out that those systems perform poorly in real-life applications [18, 19], mainly because of poor results from database querying caused by inaccurate dialog states. The work in [19] proposes to prepend the whole local knowledge base to the context, instead of doing database query. Their method achieves competent performance; However, it is not scalable with large knowledge bases. Our method makes the contribution that we introduce a knowledge retriever into the system to improve the quality of knowledge selection.

2.2. Knowledge Retriever for Conditional Generation

Recent researches such as RAG [10] and REALM [20] have introduced knowledge retrieval models into conditional generation, which greatly improves the quality of generated responses in knowledge-intensive tasks such as open-domain question answering and knowledge-grounded dialog systems. Those works use BM25 or neural network based retrievers such as DPR [21] to retrieve relevant knowledge given context and generate the answer using the retrieved knowledge pieces. There are several recent studies that improve over the original retrieval-augmented generation systems. Poly-encoder [22] proposes to use a poly-encoder instead of the original dual encoder to improve the retrieval accuracy. Fusion-in-Decoder [23] proposes to process retrieved passages independently in the encoder, but jointly fused them in the decoder, which improves the generation quality. Seeker [24] and BlenderBot3 [25] recently propose to train a model to generate better query for retrieving knowledge. However, none of these previous works apply a retrieval model to TOD systems. Our work makes the contribution that we introduce a knowledge retriever into TOD system to substitute the original dialog state tracking and database query modules, which improves the accuracy of knowledge selection.

2.3. Semi-Supervised Learning (SSL) for TOD Systems

There are increasing interests in developing SSL methods for TOD systems, which aims to leverage both labeled and unlabeled data. Latent variable modeling is an important SSL approach, which, when applied to semi-supervised TOD systems, has been

proposed in [26] and developed in [7, 27]. The dialog states and the system acts are treated as latent variables in unlabeled data and the variational algorithm is used for model training. Recently, it is shown that JSA learning significantly outperforms variational learning for semi-supervised performances [14]. Our work is different from previous works in that labels of relevant knowledge in KRTOD systems are modeled as latent variables.

3. Method

3.1. Knowledge-Retrivel TOD Model (KRTOD)

Assume we have a dialog with T turns of user utterances and system responses, denoted by $u_1, r_1, \dots, u_T, r_T$ respectively. At turn t , based on dialog context, the system queries a task-related knowledge base (KB) to get relevant knowledge, decides its action, and generates appropriate responses. The KB is composed of entities with attributes, or say, slot-value pairs, denoted by $\{sv^1, sv^2, \dots, sv^N\}$.

The slot-value pairs that are relevant for the system to respond at turn t are denoted by ξ_t . In labeled data, ξ_t is observed, while in unlabeled dialogs, it becomes a latent variable. The system action a_t could be modeled in the same way. So we denote the latent variables at turn t collectively by $h_t = \{\xi_t, a_t\}$. Similar to [14], a latent state TOD model can be defined as follows, with model parameter θ :

$$p_\theta(h_{1:T}, r_{1:T} | u_{1:T}) = \prod_{t=1}^T p_\theta(h_t, r_t | c_t, u_t) \quad (1)$$

where $c_t = u_1, r_1, \dots, u_{t-1}, r_{t-1}$ denotes the dialog context at turn t . The joint model of h_t, r_t is further decomposed into a knowledge retriever p_θ^{ret} and a response generator p_θ^{gen} .

$$p_\theta(h_t, r_t | c_t, u_t) = p_\theta^{\text{ret}}(\xi_t | c_t, u_t) \times p_\theta^{\text{gen}}(a_t, r_t | c_t, u_t, \xi_t) \quad (2)$$

From Eq. 2, we can clearly see the differences between KRTOD and traditional TOD systems. First, a retrieval model p_θ^{ret} is introduced in KRTOD to do knowledge selection procedure, instead of doing traditional database query. Second, traditional dialog state tracking is avoided as shown in Eq. 2.

In order to perform unsupervised learning over unlabeled dialogs (to be detailed below), we introduce an inference model $q_\phi(h_{1:T} | u_{1:T}, r_{1:T})$ as follows to approximate the true posterior $p_\theta(h_{1:T} | u_{1:T}, r_{1:T})$:

$$\begin{aligned} q_\phi(h_{1:T} | u_{1:T}, r_{1:T}) &= \prod_{t=1}^T q_\phi(h_t | c_t, u_t, r_t) \\ &= \prod_{t=1}^T q_\phi(\xi_t, a_t | c_t, u_t, r_t) \end{aligned} \quad (3)$$

3.2. Model Implementation

To implement the models introduced in Section 3.1, we use a BERT [28] based classification model to build the knowledge retriever p_θ^{ret} and GPT2 based autoregressive generative models to build both the generation model p_θ^{gen} and the inference model q_ϕ , as illustrated in Figure 2.

The retrieval model p_θ^{ret} aims to retrieve relevant knowledge from the KB in order to respond to user utterances. Particularly, the knowledge piece ξ_t necessary for turn t is represented by $\xi_t \triangleq \xi_t^1 \oplus \xi_t^2 \oplus \dots \oplus \xi_t^N$, where \oplus denotes sequence concatenation. $\xi_t^i = sv^i$ if the slot-value sv^i is relevant to the response r_t ; otherwise, ξ_t^i is set to be empty. Moreover, we assume that whether sv^i is contained in ξ_t or not is independent of each other. Therefore, the retrieval probability can be written as follows:

$$p_\theta^{\text{ret}}(\xi_t | c_t, u_t) = \prod_{i=1}^N p_\theta^{\text{ret}}(\xi_t^i | c_t, u_t) \quad (4)$$

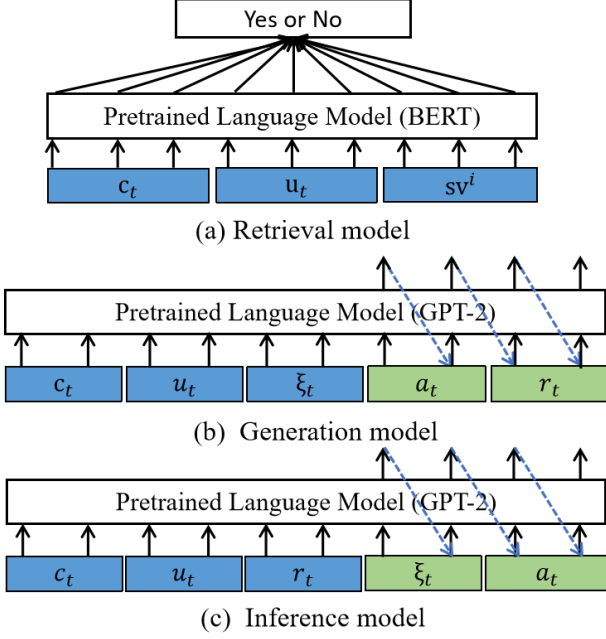


Figure 2: The model implementations: (a) the retrieval model, (b) the generation model, and (c) the inference model for our JSA-KRTOD. All of the variables $c_t, u_t, \xi_t, a_t, r_t, sv^i$ are represented by token sequences in our experiments.

where $p_{\theta}^{\text{ret}}(\xi_t^i | c_t, u_t)$ is realized based on BERT, using $c_t \oplus u_t \oplus sv^i$ as input, as shown in Figure 2. Specifically, denote the mean-pooling of the BERT encoder output by

$$\mathbf{x} = \text{meanpooling}(\text{BERT}(c_t \oplus u_t \oplus sv^i))$$

Then we have

$$p_{\theta}^{\text{ret}}(\xi_t^i = sv^i | c_t, u_t) = \frac{1}{1 + e^{-(\mathbf{W}\mathbf{x} + b)}}$$

$$p_{\theta}^{\text{ret}}(\xi_t^i = \text{NULL} | c_t, u_t) = 1 - p_{\theta}^{\text{ret}}(\xi_t^i = sv^i | c_t, u_t)$$

where \mathbf{W}, b are trainable parameters.

As shown in Figure 2, both the generation model and the inference model are realized based on GPT2, and the probabilities are calculated in an autoregressive way as follows:

$$p_{\theta}^{\text{gen}}(a_t, r_t | c_t, u_t, \xi_t)$$

$$= \prod_{l=1}^{|\xi_t \oplus r_t|} p_{\theta}^{\text{gen}}(y^l | c_t, u_t, \xi_t, y^1, \dots, y^{l-1}) \quad (5)$$

$$q_{\phi}(\xi_t, a_t | c_t, u_t, r_t)$$

$$= \prod_{l=1}^{|\xi_t \oplus a_t|} q_{\phi}(y^l | c_t, u_t, r_t, y^1, \dots, y^{l-1}) \quad (6)$$

where $|\cdot|$ denotes the length in tokens, and y^l the l -th token.

In supervised training, we use the ground truth ξ_t , which is annotated in the dataset, to maximize the log probabilities in Eq. 4 - 6. In testing, according to Eq. 2, we firstly retrieve relevant slot-value pairs ξ_t by thresholding $p_{\theta}^{\text{ret}}(\xi_t^i = sv^i | c_t, u_t), i = 1, \dots, N$; then, we generate a_t and r_t , based on retrieved ξ_t .

3.3. Semi-Supervised Training of KRTOD via JSA

Consider an unlabeled dialog of T turns, where the user utterances and system responses $u_{1:T}, r_{1:T}$ are given, but there are

Algorithm 1 Semi-supervised training in JSA-KRTOD

Input: A mix of labeled and unlabeled dialogs.

- 1: Run supervised pre-training of θ and ϕ on labeled dialogs;
- 2: **repeat**
- 3: Draw a dialog $(u_{1:T}, r_{1:T})$;
- 4: **if** $(u_{1:T}, r_{1:T})$ is not labeled **then**
- 5: Generate $h_{1:T}$ using the recursive turn-level MIS sampler
- 6: **end if**
- 7: $J_{\theta} = 0, J_{\phi} = 0$;
- 8: **for** $i = 1, \dots, T$ **do**
- 9: $J_{\theta} + = \log p_{\theta}^{\text{gen}}(a_t, r_t | c_t, u_t, \xi_t)$;
- 10: $J_{\phi} + = \log q_{\phi}(\xi_t, a_t | c_t, u_t, r_t)$;
- 11: **end for**
- 12: Update θ by ascending: $\nabla_{\theta} J_{\theta}$;
- 13: Update ϕ by ascending: $\nabla_{\phi} J_{\phi}$;
- 14: **until** convergence
- 15: **return** θ and ϕ

no labels for the KB results and system acts, thus being treated as latent variables $h_{1:T} \triangleq \{\xi_{1:T}, a_{1:T}\}$. We can use JSA learning to maximize marginal likelihood $p_{\theta}(r_{1:T} | u_{1:T})$, which iterates Monte Carlo sampling and parameter updating. Particularly, we use a recursive turn-level MIS sampler to sample $h_{1:T}$, as developed in [14]. At each turn t , the MIS sampler works in a propose, accept or reject way, as follows:

- 1) Propose $h'_t \sim q_{\phi}(h_t | c_t, u_t, r_t)$.
- 2) Simulate $\eta \sim \text{Uniform}[0, 1]$ and let

$$h_t = \begin{cases} h'_t, & \text{if } \eta \leq \min \left\{ 1, \frac{w(h'_t)}{w(\bar{h}_t)} \right\} \\ \bar{h}_t, & \text{otherwise} \end{cases} \quad (7)$$

where \bar{h}_t denotes the cached latent state, and the importance ratio $w(h_t)$ between the target and the proposal distribution can be written as:

$$w(h_t) \propto \frac{p_{\theta}(h_t, r_t | c_t, u_t)}{q_{\phi}(h_t | c_t, u_t, r_t)}$$

$$= \frac{p_{\theta}^{\text{ret}}(\xi_t | c_t, u_t) \times p_{\theta}^{\text{gen}}(a_t, r_t | c_t, u_t, \xi_t)}{q_{\phi}(\xi_t, a_t | c_t, u_t, r_t)} \quad (8)$$

Now that we have introduced the method of how to deal with unlabeled data in JSA learning. Semi-supervised learning over a mix of labeled and unlabeled data could be readily realized in JSA-KRTOD by maximizing the sum of $\log p_{\theta}(h_{1:T}, r_{1:T} | u_{1:T})$ (the conditional joint log-likelihood) over labeled data and $\log p_{\theta}(r_{1:T} | u_{1:T})$ (the conditional marginal log-likelihood) over unlabeled data.

The semi-supervised training procedure of JSA-KRTOD is summarized in Algorithm 1. Specifically, we first conduct supervised pre-training of both the generative model p_{θ}^{gen} and the inference model q_{ϕ} on labeled data.

Then we randomly draw supervised and unsupervised mini-batches from labeled and unlabeled data. For labeled dialogs, the latent states $h_t \triangleq \{\xi_t, a_t\}$ are given. For unlabeled dialogs, we apply the recursive turn-level MIS sampler to sample the latent states h_t and treat them as if being given. The gradients calculation and parameter updating are then the same for labeled and unlabeled dialogs.¹

¹Only the response generator p_{θ}^{gen} is updated in Line 9 in Algorithm 1. After supervised pre-training, the retrieval model p_{θ}^{ret} is not updated over unlabeled data, since the KB is not available for unlabeled data.

4. Experiments

4.1. Experiment Settings and Baselines

Experiments are conducted on a real-life human-human dialog dataset, called MobileCS, released from the EMNLP 2022 SereTOD Challenge [15]. The MobileCS dataset is from customer-service logs, instead of collected by the Wizard-of-Oz method [9]. Therefore, building TOD systems over such dataset is more challenging, as user utterances are more casual and may contain noises from automatic speech recognition [15]. MobileCS contains a total of around 100K dialogs. The labeled part was officially randomly split into train, development, and test sets, which consist of 8,953, 1014 and 955 dialog samples, respectively. The remaining 87,933 dialogs are unlabeled. Therefore, experiments in both labeled-only and semi-supervised settings (over both labeled and unlabeled data) can be conducted and fairly compared. For evaluation, we follow the original scripts in [18] and mainly focus on two metrics, *Success rate* and *BLEU*, which mainly evaluate the quality of the generated responses. *Success rate* measures how often the system is able to provide all the entities and values requested by the user, which is crucial in performing a successful dialog. *BLEU* is used to measure the fluency of the generated responses by analyzing the amount of n-gram overlap between the real responses and the generations. The overall performance is measured by *Combined score*, which is $Success + 2 * BLEU$, as in the original scripts. Several strong baselines are compared with our method. The official baseline [18] uses predicted dialog state to query the database, which is referred to as KB-query in Table 1. PRIS [19] concatenates the whole local KB to the dialog history, which is referred to as KB-grounded in Table 1. TJU-LMC [29] uses coarse-to-fine intent detection to improve performance over the baseline, while Passion [30] improves prompting scheme to boost performance. PRIS, Passion, and TJU-LMC were top three teams in the SereTOD Challenge. In our experiments, to follow the Challenge guideline, hyper-parameters are chosen based on the development set, and evaluated on the test set.

4.2. Main Results

As shown in Table 1, our method achieves SOTA results on MobileCS in both labeled-only and semi-supervised settings. The results clearly show the superiority of our method, which greatly outperforms the baseline method (KB-query) and the competitive KB-grounded method, especially in the Success rate metric. It is evident that importing a knowledge retriever into a TOD system can significantly enhance the system’s ability in accurately providing the necessary knowledge for the current turn, which is the key to achieve high Success rate.

From the comparison between the semi-supervised and the labeled-only results in Table 1, we can see that performing semi-supervision using unlabeled data is crucial for improving the system’s performance. Using the proposed JSA-KRTOD to perform semi-supervision can bring substantial gain in both the Success rate and BLEU-4 metrics, and improves over the strong KRTOD baseline by **7.07** in the Combined score.

It can be seen from Table 1 that although our model achieves the SOTA results on Combined score and Success rate, the BLEU-4 score is not so competitive. Presumably, the main reason for the difference in the BLEU-4 performance lies in the number of model parameters. Larger model (T5 1B) can better fit the dataset, resulting higher BLEU score. The comparison between the KB-grounded method and our proposed method using the same model (GPT2 100M) in the Labeled-only part of Table 1 shows that the two methods have similar BLEU scores, while our method improves greatly on the Success rate. Also

Table 1: *Main results on the MobileCS dataset. Success, BLEU-4, and Combined score are reported. Our approach achieves SOTA results on both labeled-only and semi-supervised settings. Within the parentheses show the backbone models and their number of parameter.*

Setting	Method	Success	BLEU-4	Combined
Semi-supervised	Baseline [18]	31.5	4.170	39.84
	Passion [30]	43.2	6.790	56.78
	TJU-LMC [29]	68.9	7.54	83.98
	PRIS [19]	78.9	14.51	107.92
	JSA-KRTOD	91.8	9.677	111.15
Labeled-only	KB-query (GPT2 100M) [18]	31.5	4.170	39.84
	KB-grounded (GPT2 100M) [19]	64.2	8.845	81.89
	KB-grounded (T5 1B) [19]	74.1	11.32	96.74
	KRTOD (GPT2 100M)	86.8	8.639	104.08

Table 2: *Comparison between pseudo labeling (PL) and JSA learning methods. Ratio means the ratio between the number of unlabeled dialogs and the number of labeled dialogs in training. p-value means the significant test result for Combined score.*

Ratio	Method	Success	BLEU-4	Combined	p-value
1:1	PL	87.5	8.853	105.21	0.589
	JSA	88.0	8.713	105.43	
2:1	PL	87.8	9.196	106.19	0.853
	JSA	88.7	9.490	107.68	
4:1	PL	88.5	9.341	107.18	0.037
	JSA	90.9	9.398	109.70	
9:1	PL	89.4	9.532	108.46	0.055
	JSA	91.8	9.677	111.15	

note that PRIS [19] uses large-scale customer-service dialogs to pre-train their model (T5 1B), which further improves the BLEU score over their GPT2 based model. Therefore, further work can be done to combine our proposed method with a larger backbone.

4.3. Analysis and Ablation

We further examine whether our semi-supervised method JSA-KRTOD is competitive among other semi-supervised methods. We mainly compare our method with a classic semi-supervised method, called pseudo labeling (PL) or self-training (ST), which has been used in [19] and [27] for TOD systems. We implement the PL method, similar to the “ST with inference model” method in [27]. The results in Table 2 show that JSA-KRTOD outperforms PL constantly in all ratios. The relative improvement of JSA over PL in reducing errors in Success rate is 23% under ratio 9:1. Further, the p-values from the matched-pairs significance tests [31] in Combined score show that as the size of unlabeled data increases, the improvements of JSA-KRTOD over PL become more significant, confirming the superiority of JSA-KRTOD in leveraging unlabeled data.

5. Conclusion and Future Work

In this paper, we propose KRTOD, which introduces a knowledge retriever into the TOD system. The knowledge retriever relieves the burden to track the dialog state and the inflexibility of using the dialog state to query the database in the traditional TOD systems. The proposed KRTOD method greatly outperforms previous strong baseline methods over the challenging real-life MobileCS dataset. Moreover, by combining KRTOD with JSA based semi-supervised learning, our semi-supervised TOD system, JSA-KRTOD, improves over the strong supervised baseline and achieves SOTA results on MobileCS. For future work, KRTOD potentially can exploit more types of knowledge sources, such as passages, documents and knowledge graphs, in addition to slot-value pairs used in this paper.

6. References

- [1] J. D. Williams, A. Raux, and M. Henderson, “The dialog state tracking challenge series: A review,” *Dialogue & Discourse*, vol. 7, no. 3, pp. 4–33, 2016.
- [2] Y. Zhang, Z. Ou, and Z. Yu, “Task-oriented dialog systems that consider multiple appropriate responses under the same context,” in *The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, 2020.
- [3] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.
- [4] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.
- [5] E. Hosseini-Asl, B. McCann, C.-S. Wu, S. Yavuz, and R. Socher, “A simple language model for task-oriented dialogue,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 20 179–20 191, 2020.
- [6] B. Peng, C. Li, J. Li, S. Shayandeh, L. Liden, and J. Gao, “Soloist: Building task bots at scale with transfer learning and machine teaching,” *Transactions of the Association for Computational Linguistics (ACL)*, 2021.
- [7] H. Liu, Y. Cai, Z. Ou, Y. Huang, and J. Feng, “Revisiting markovian generative architectures for efficient task-oriented dialog systems,” in *IEEE Spoken Language Technology Workshop*, 2022.
- [8] Y. Lee, “Improving end-to-end task-oriented dialog system with a simple auxiliary task,” in *Findings of the Association for Computational Linguistics: EMNLP*, 2021, pp. 1296–1303.
- [9] P. Budzianowski, T.-H. Wen, B.-H. Tseng, I. Casanueva, U. Stefan, R. Osman, and M. Gašić, “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 (EMNLP)*, 2018.
- [10] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, “Retrieval-augmented generation for knowledge-intensive nlp tasks,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.
- [11] M. Kang, J. M. Kwak, J. Baek, and S. J. Hwang, “Knowledge-consistent dialogue generation with knowledge graphs,” in *ICML 2022 Workshop on Knowledge Retrieval and Language Models*.
- [12] Y. Qu, Y. Ding, J. Liu, K. Liu, R. Ren, W. X. Zhao, D. Dong, H. Wu, and H. Wang, “Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering,” in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 5835–5847.
- [13] Z. Ou and Y. Song, “Joint stochastic approximation and its application to learning discrete latent variable models,” in *Conference on Uncertainty in Artificial Intelligence*, 2020, pp. 929–938.
- [14] Y. Cai, H. Liu, Z. Ou, Y. Huang, and J. Feng, “Advancing semi-supervised task oriented dialog systems by JSA learning of discrete latent variable models,” in *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2022, pp. 456–467.
- [15] Z. Ou, J. Feng, J. Li, Y. Li, H. Liu, H. Peng, Y. Huang, and J. Zhao, “A challenge on semi-supervised and reinforced task-oriented dialog systems,” *arXiv preprint arXiv:2207.02657*, 2022.
- [16] M. Heck, C. van Niekerk, N. Lubis, C. Geishausser, H.-C. Lin, M. Moresi, and M. Gasic, “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*, 2020.
- [17] C.-H. Lee, H. Cheng, and M. Ostendorf, “Dialogue state tracking with a language model using schema-driven prompting,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021, pp. 4937–4949.
- [18] H. Liu, H. Peng, Z. Ou, J. Li, Y. Huang, and J. Feng, “Information extraction and human-robot dialogue towards real-life tasks: A baseline study with the mobilecscs dataset,” in *EMNLP 2022 SereTOD Workshop*, 2022.
- [19] W. Zeng, K. He, Z. Wang, D. Fu, G. Dong, R. Geng, P. Wang, J. Wang, C. Sun, W. Wu *et al.*, “Semi-supervised knowledge-grounded pre-training for task-oriented dialog systems,” in *EMNLP 2022 SereTOD Workshop*, 2022.
- [20] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang, “Realm: retrieval-augmented language model pre-training,” in *Proceedings of the 37th International Conference on Machine Learning*, 2020, pp. 3929–3938.
- [21] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, “Dense passage retrieval for open-domain question answering,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 6769–6781.
- [22] S. Humeau, K. Shuster, M.-A. Lachaux, and J. Weston, “Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring,” in *International Conference on Learning Representations*, 2020.
- [23] G. Izacard and É. Grave, “Leveraging passage retrieval with generative models for open domain question answering,” in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2021, pp. 874–880.
- [24] K. Shuster, M. Komeili, L. Adolphs, S. Roller, A. Szlam, and J. Weston, “Language models that seek for knowledge: Modular search & generation for dialogue and prompt completion,” *arXiv preprint arXiv:2203.13224*, 2022.
- [25] K. Shuster, J. Xu, M. Komeili, D. Ju, E. M. Smith, S. Roller, M. Ung, M. Chen, K. Arora, J. Lane *et al.*, “Blenderbot 3: a deployed conversational agent that continually learns to responsibly engage,” *arXiv preprint arXiv:2208.03188*, 2022.
- [26] Y. Zhang, Z. Ou, M. Hu, and J. Feng, “A probabilistic end-to-end task-oriented dialog model with latent belief states towards semi-supervised learning,” in *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.
- [27] H. Liu, Y. Cai, Z. Lin, Z. Ou, Y. Huang, and J. Feng, “Variational latent-state GPT for semi-supervised task-oriented dialog systems,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.
- [28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019.
- [29] Z. Yang, X. Ma, A. Liu, and Z. Zhang, “Discovering customer-service dialog system with semi-supervised learning and coarse-to-fine intent detection,” in *EMNLP 2022 SereTOD Workshop*, 2022.
- [30] W. Lu, Y. Wang, W. Zhao, B. Chen, X. Chen, J. Pan, W. Liang, and Y. Lai, “Team passion at seretod-emnlp 2022: End-to-end task-oriented dialog system with improved prompting scheme,” 2022. [Online]. Available: <http://seretod.org/Challenge.html>
- [31] L. Gillick and S. J. Cox, “Some statistical issues in the comparison of speech recognition algorithms,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1989.