End-to-End Optimization of Task-Oriented Dialogue Model with Deep Reinforcement Learning

Bing Liu[1]*, **Gokhan Tür**[2], **Dilek Hakkani-Tür**[2], **Pararth Shah**[2], **Larry Heck**[2]
[1]Carnegie Mellon University, Pittsburgh, PA 15213
[2]Google Research, Mountain View, CA 94043
liubing@cmu.edu, {gokhant,dilekh,pararth,larryheck}@google.com

Abstract

In this paper, we present a neural network based task-oriented dialogue system that can be optimized end-to-end with deep reinforcement learning (RL). The system is able to track dialogue state, interface with knowledge bases, and incorporate query results into agent's responses to successfully complete task-oriented dialogues. Dialogue policy learning is conducted with a hybrid supervised and deep RL methods. We first train the dialogue agent in a supervised manner by learning directly from task-oriented dialogue corpora, and further optimize it with deep RL during its interaction with users. In the experiments on two different dialogue task domains, our model demonstrates robust performance in tracking dialogue state and producing reasonable system responses. We show that deep RL based optimization leads to significant improvement on task success rate and reduction in dialogue length comparing to supervised training model. We further show benefits of training task-oriented dialogue model end-to-end comparing to component-wise optimization with experiment results on dialogue simulations and human evaluations.

1 Introduction

Task-oriented dialogue, different from chit-chat type of conversation, requires the system to produce responses by accessing information from knowledge bases and planning over multiple dialogue turns. Conventional task-oriented dialogue systems have a complex pipeline [1, 2] consisting of independently developed and modularly connected components for natural language understanding (NLU) [3, 4], dialogue state tracking (DST) [5, 6], and dialogue policy [7, 8]. A limitation with such pipelined design is that errors made in upper stream modules may propagate to downstream components, making it hard to identify and track the source of errors. Moreover, each component in the pipeline is ideally re-trained as preceding components are updated, so that we have inputs similar to the training examples at run-time. This domino effect causes several issues in practice.

To ameliorate these limitations with the conventional pipeline dialogue systems, recent efforts have been made in designing neural network based end-to-end learning solutions. Such end-to-end systems aim to optimize directly towards final system objectives (e.g. response generation, task success rate) instead of performing component-wise optimization. Many of the recently proposed end-to-end models are trained in supervised manner [9, 10, 11, 12] by learning from human-human or human-machine dialogue corpora. Deep reinforcement learning (RL) based systems [13, 14, 15, 16] that learns by interacting with human user or user simulator have also been studied in the literature. Comparing to supervised training models, systems trained with deep RL showed improved task success rate and model robustness towards diverse dialogue scenarios.

*Work done while the author was an intern at Google Research.

In this work, we present a neural network based task-oriented dialogue system that can be optimized end-to-end with deep RL. The system is built with neural network components for natural language encoding, dialogue state tracking, and dialogue policy learning. Each system component takes in underlying component's outputs in a continuous from which is fully differentiable with respect to the system optimization target, and thus the entire system can be trained end-to-end. In the experiments on a movie booking domain, we show that our system trained with deep RL leads to significant improvement on dialogue task success rate comparing to supervised training systems. We further illustrate the benefit of performing end-to-end optimization comparing to only updating the policy network during online policy learning as in many previous work [7, 8].

2 Related Work

Traditional task-oriented dialogue systems typically require a large number of handcrafted features, making it hard to extend a system to new application domains. Recent approaches to task-oriented dialogue treat the task as a partially observable Markov Decision Process (POMDP) [2] and use RL for online policy optimization by interacting with users [17]. The dialogue state and action space have to be carefully designed in order to make the reinforcement policy learning tractable [2].

With the success of end-to-end trainable neural network models in modeling non-task-oriented chit-chat dialogues [18, 19], efforts have been made in carrying over the good performance of end-to-end models to task-oriented dialogues. Bordes and Weston [10] proposed modeling task-oriented dialogues with a machine reading approach using end-to-end memory networks. Their model removes the dialogue state tracking module and selects the final system response directly from candidate responses. Comparing to this approach, our model explicitly tracks user's goal in dialogue state over the sequence of turns, as robust dialogue state tracking has been shown [20, 16] to be useful for interfacing with a knowledge base (KB) and improving task success rate. Wen et al. [9] proposed an end-to-end trainable neural network model with modularly connected system components. This system is trained in a supervised manner, and thus may not be robust enough to handle diverse dialogue situations due to the limited varieties in the training dialogue samples. Our system is trained by a combination of SL and deep RL methods, as it is shown that RL training may effectively improve the system robustness and dialogue success rate [13, 15]. Dhingra et al. [16] proposed an end-to-end RL dialogue agent for information access. Their model focuses on bringing differentiability to the KB query operation by introducing a "soft" retrieval process in selecting the KB entries. Such soft-KB lookup may be prone to information updates in the KB, which is common in real world information systems. In our model, we use symbolic query and leave the selection of KB entities to external services (e.g. a recommender system), as entity ranking in real world systems can be made with much richer feature sets (e.g. user profiles, location and time context, etc.). Quality of the generated query is directly related to the performance of our dialog state tracking module, which can be optimized during user interactions in the proposed end-to-end reinforcement learning model.

3 Proposed Method

3.1 System Architecture

Figure 1 shows the overall system architecture of the proposed end-to-end task-oriented dialogue model. A continuous form dialogue state over a sequence of turns is maintained in the state s_k of a dialogue-level LSTM. At each dialogue turn k, this dialogue-level LSTM takes in the encoding of the user utterance U_k and the encoding of the previous system action A_{k-1}, and produces a probability distribution $P(l_k^m)$ over candidate values for each of the tracked goal slots:

$$s_k = \text{LSTM}(s_{k-1},\ [U_k,\ A_{k-1}]) \tag{1}$$

$$P(l_k^m \mid \mathbf{U}_{\leq k},\ \mathbf{A}_{<k}) = \text{SlotDist}_m(s_k) \tag{2}$$

where SlotDist_m is a single hidden layer MLP with softmax activation function over slot type $m \in M$. In encoding natural language user utterance to a continuous vector U_k, we use a bidirectional LSTM (i.e. an utterance-level LSTM) reader by concatenating the last forward and backward LSTM states.

Based on slot-value pair outputs from dialogue state tracking, a query command is formulated by filling a query template with candidate values that have the highest probability for each tracked goal

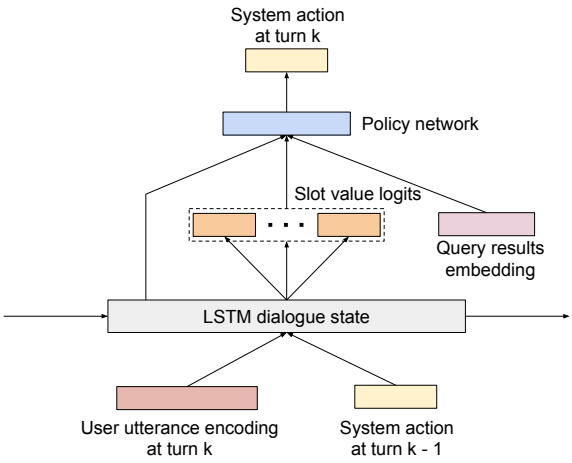


Figure 1: Proposed end-to-end task-oriented dialogue model architecture.

slot. Alternatively, an n-best list of queries can be generated with the most probable candidate values. The query is sent to a KB to retrieve user requested information. Finally, a system action is emitted in response to the user's input based on the current dialogue state and the information retrieved from the knowledge base:

$$P(a_k \mid U_{\leq k},\ A_{<k},\ E_{\leq k}) = \mathrm{PolicyNet}(s_k, v_k, E_k) \tag{3}$$

where v_k represents the concatenated log probabilities of candidate values for each goal slot. E_k is the encoding of the retrieved result from the knowledge base (e.g. item availability and number of matched items). $\mathrm{PolicyNet}$ is an MLP with $\mathrm{softmax}$ activation function over all system actions. The emitted system action is then translated to a system response in natural language format by combining the state tracking outputs and the query results. We use a template based natural language generator (NLG) in this work.

3.2 Model Training

We first train the system in a supervised manner using task-oriented dialogue corpora. Based on system inputs with past user utterances, system actions, and KB results, the model tracks the user's goal slot values and predict the next system action. We optimize the model to minimize the linear interpolation of cross-entropy losses for dialogue state tracking and system action prediction:

$$\begin{aligned}\min_{\theta} \sum_{k=1}^{K} -\Big[\sum_{m=1}^{M} \lambda_{l^m} \log P(l_k^{m*} | \mathbf{U}_{\leq k}, \mathbf{A}_{<k}, \mathbf{E}_{<k}; \theta) \\ + \lambda_a \log P(a_k^* | \mathbf{U}_{\leq k}, \mathbf{A}_{<k}, \mathbf{E}_{\leq k}; \theta) \Big]\end{aligned} \tag{4}$$

where λs are the linear interpolation weights for the cost of each system output. l_k^{m*} and a_k^* are the ground truth labels for goal slots and system action the kth turn.

After the supervised training stage, we further optimize the system with RL by letting the agent to interact with users and collecting user feedback. We apply REINFORCE algorithm [21] in optimizing the network parameters. We use softmax policy during RL training to encourage the agent to explore the dialogue action space. Feedback is only collected at the end of a dialogue. A positive reward is assigned for success tasks, and a zero reward is assigned for failure tasks. A small step penalty is applied to each dialogue turn to encourage the agent to complete the task in fewer steps. We use policy gradient method for dialogue policy learning. With likelihood ratio gradient estimator, the gradient of the objective function $J_k(\theta)$ can be derived as:

$$\nabla_\theta J_k(\theta) = \nabla_\theta \mathbb{E}_\theta \left[R_k\right] = \mathbb{E}_{\theta_a} \left[\nabla_\theta \log \pi_\theta(a_k|s_k) R_k\right] \tag{5}$$

This last expression above gives us an unbiased gradient estimator. We sample the agent action based on the currently learned policy at each dialogue turn and compute the gradient.

4 Experiments

4.1 Datasets

We evaluate the proposed method on DSTC2 [22] dataset in restaurant search domain and an internally collected dialogue corpus in movie booking domain. The movie booking corpus is generated with rule based dialogue agent and user simulator. The same user simulator is used to interact with our end-to-end learning agent during RL training. We use an extended set of NLG templates during model testing to evaluate the end-to-end model's generalization capability in handling diverse natural language inputs.

4.2 Training Settings

We set state size of the dialogue-level and utterance-level LSTM as 200 and 150 respectively. Hidden layer size of the policy network is set as 100. We used randomly initialized word embedding of size 300. Adam optimization method [23] with initial learning rate of 1e-3 is used for mini-batch training. Dropout rate of 0.5 is applied during training to prevent the model from over-fitting.

In dialogue simulation, we take a task-oriented dialogue as successful if the goal slot values estimated by the state tracker fully match to the user's true goal values, and the system is able to offer an entity which is finally accepted by the user. Maximum allowed number of dialogue turn is set as 15. A positive reward of +15.0 is given to the agent at the end of a success dialogue, and a zero reward is given in a failure case. We apply a step penalty of -1.0 for each turn to encourage shorter dialogue in completing the task.

4.3 Results and Analysis

Table 1 and Table 2 show the supervised training model performance on DSTC2 and the movie booking dialogue dataset. The model is evaluated on dialogue state tracking accuracy. On DSTC2 dataset, our end-to-end model achieves near-state-of-the-art state tracking performance comparing to the recent published results using RNN [24] and NBT [6]. On the movie booking dataset, our model also achieves promising performance on individual slot tracking and joint slot tracking accuracy.

Table 1: Belief tracking results on DSTC2 corpus (with ASR hypothesis as input)

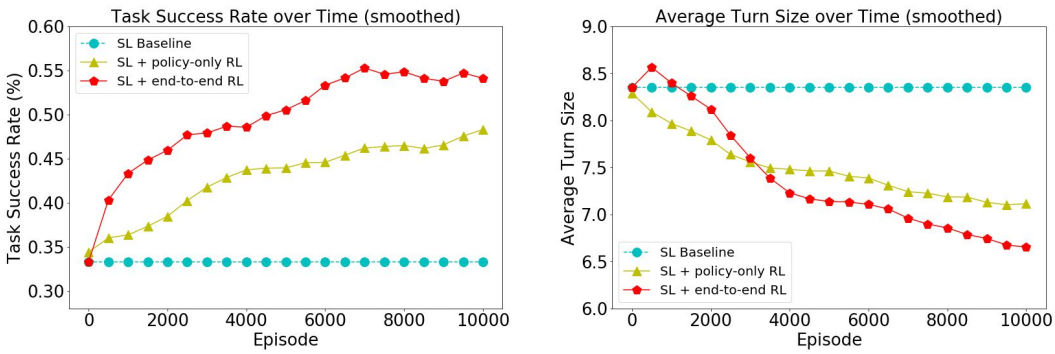
Model	**Area**	**Food**	**Price**	**Joint**
RNN [24]	92	86	86	69
NBT [6]	90	84	94	72
Our end-to-end model	90	84	92	72

Table 2: Belief tracking results on movie booking dataset

Model	**Num_ticket**	**Movie**	**Theater**	**Date**	**Time**	**Joint**
Our end-to-end model	98.22	91.86	97.33	99.31	97.71	84.57

Figure 2 shows the RL curves of the proposed model on dialogue task success rate and average dialogue turn size. Evaluation is based on dialogue simulations between our proposed end-to-end dialogue agent and the rule based user simulator. This is different from the evaluations based on fixed dialogue corpora as in Table 1 and 2. The policy gradient based RL training is performed on top of the supervised training model. We compare models with two RL training settings, the end-to-end training and the policy-only training, to the baseline supervised learning (SL) model.

As shown in Figure 2(a), the SL model performs poorly during user interaction, indicating the limited generalization capability of the SL model to unseen dialogue state. Any mistake made by the agent during user interaction may lead to deviation of the dialogue from the training dialogue trajectories and states. The SL agent does not know how to recover from an unknown state, which leads to final task failure. RL model training, under both end-to-end learning and policy-only learning settings, continuously improves the task success rate with the growing number of user interactions. We see clear advantage of performing end-to-end model update in achieving higher dialogue task success rate comparing to only updating the policy network during interactive learning.



(a) (b)

Figure 2: RL curves on (a) dialogue task success rate and (b) average dialogue turn size.

Figure 2(b) shows the learning curves for the average number of turns in successful dialogues. We observe decreasing number of dialogue turns along the growing number of interactive learning episodes. This shows that the dialogue agent learns better strategies to successfully complete the task in fewer numbers of turns. Similar to the results for task success rate, the end-to-end training model outperforms the model with policy-only optimization during RL training, achieving lower average number of dialogue turns in successfully completing a task.

4.4 Human Evaluations

We further evaluate our proposed method with human judges recruited via Amazon Mechanical Turk. Each judge is asked to read a dialogue between our model and the user simulator and rate each system turn on a scale of 1 (frustrating) to 5 (optimal way to help the user). Each turn is rated by 3 different judges. We rate the three models with 100 dialogues each: (i) the SL model, (ii) SL with policy-only RL model, and (iii) SL with end-to-end RL model. Table 3 lists the mean and standard deviation of human evaluation scores over all system turns: end-to-end optimization with RL clearly improves the quality of the model according to human judges.

Table 3: Human evaluation results with mean and standard deviation of crowd worker scores.

Model	SL	SL + policy-only RL	SL + end-to-end RL
Score	3.987 ± 0.086	4.261 ± 0.089	4.394 ± 0.087

5 Conclusions

In this work, we propose a neural network based task-oriented dialogue system that can be trained end-to-end with supervised learning and deep reinforcement learning. We first bootstrap a dialogue agent with supervised training by learning directly from task-oriented dialogue corpora, and further optimize it with deep RL during its interaction with users. We show in the experiments that deep RL optimization on top of the supervised training model leads to significant improvement on task success rate and reduction in dialogue length comparing to supervised training baseline model. The simulation and human evaluation results further illustrate benefits of performing end-to-end model training with deep RL comparing to component-wise optimization.

References

[1] Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. Let's go public! taking a spoken dialog system to the real world. In *in Proc. of Interspeech 2005*, 2005.

[2] Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 2013.

[3] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, et al. Using recurrent neural networks for slot filling in spoken language understanding. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 2015.

[4] Bing Liu and Ian Lane. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Interspeech 2016*, 2016.

[5] Matthew Henderson, Blaise Thomson, and Steve Young. Word-based dialog state tracking with recurrent neural networks. In *SIGDIAL*, pages 292–299, 2014.

[6] Nikola Mrkšić, Diarmuid O Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. Neural belief tracker: Data-driven dialogue state tracking. *arXiv preprint arXiv:1606.03777*, 2016.

[7] Milica Gasic and Steve Young. Gaussian processes for pomdp-based dialogue manager optimization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2014.

[8] Pei-Hao Su, Milica Gasic, Nikola Mrksic, Lina Rojas-Barahona, Stefan Ultes, David Vandyke, Tsung-Hsien Wen, and Steve Young. On-line active reward learning for policy optimisation in spoken dialogue systems. In *ACL*, 2016.

[9] Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. A network-based end-to-end trainable task-oriented dialogue system. In *EACL*, 2017.

[10] Antoine Bordes and Jason Weston. Learning end-to-end goal-oriented dialog. In *ICLR*, 2017.

[11] Mihail Eric and Christopher D Manning. A copy-augmented sequence-to-sequence architecture gives good performance on task-oriented dialogue. *arXiv preprint arXiv:1701.04024*, 2017.

[12] Bing Liu and Ian Lane. An end-to-end trainable neural network model with belief tracking for task-oriented dialog. In *Interspeech*, 2017.

[13] Xuijun Li, Yun-Nung Chen, Lihong Li, and Jianfeng Gao. End-to-end task-completion neural dialogue systems. *arXiv preprint arXiv:1703.01008*, 2017.

[14] Bing Liu and Ian Lane. Iterative policy learning in end-to-end trainable task-oriented neural dialog models. In *ASRU*, 2017.

[15] Jason D Williams, Kavosh Asadi, and Geoffrey Zweig. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *ACL*, 2017.

[16] Bhuwan Dhingra, Lihong Li, Xiujun Li, Jianfeng Gao, Yun-Nung Chen, Faisal Ahmed, and Li Deng. Towards end-to-end reinforcement learning of dialogue agents for information access. In *Proceedings of ACL*, 2017.

[17] M Gašić, Catherine Breslin, Matthew Henderson, Dongho Kim, Martin Szummer, Blaise Thomson, Pirros Tsiakoulis, and Steve Young. On-line policy optimisation of bayesian spoken dialogue systems via human interaction. In *ICASSP*, 2013.

[18] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. *arXiv preprint arXiv:1507.04808*, 2015.

[19] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. In *EMNLP*, 2016.

[20] Filip Jurčíček, Blaise Thomson, and Steve Young. Reinforcement learning for parameter estimation in statistical spoken dialogue systems. *Computer Speech & Language*, 2012.

[21] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 1992.

[22] Matthew Henderson, Blaise Thomson, and Jason Williams. The second dialog state tracking challenge. In *SIGDIAL*, 2014.

[23] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014.

[24] Matthew Henderson, Blaise Thomson, and Steve Young. Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised gate. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, 2014.