Amalgamating Knowledge from Two Teachers for Task-oriented Dialogue System with Adversarial Training

Wanwei He[1,2], Min Yang[2]*, Rui Yan[3], Chengming Li[2], Ying Shen[4], Ruifeng Xu[5]
[1]University of Chinese Academy of Sciences, China
[2]Shenzhen Key Laboratory for High Performance Data Mining,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[3]Wangxuan Institute of Computer Technology, Peking University, China
[4] School of Intelligent Systems Engineering, Sun Yat-Sen University, China
[5]Harbin Institute of Technology (Shenzhen), China
{ww.he, min.yang, cm.li}@siat.ac.cn, ruiyan@pku.edu.cn
sheny76@mail.sysu.edu.cn, xuruifeng@hit.edu.cn

Abstract

The challenge of both achieving task completion by querying the knowledge base and generating human-like responses for task-oriented dialogue systems is attracting increasing research attention. In this paper, we propose a "Two-Teacher One-Student" learning framework (TTOS) for task-oriented dialogue, with the goal of retrieving accurate KB entities and generating human-like responses simultaneously. TTOS amalgamates knowledge from two teacher networks that together provide comprehensive guidance to build a high-quality task-oriented dialogue system (student network). Each teacher network is trained via reinforcement learning with a goal-specific reward, which can be viewed as an expert towards the goal and transfers the professional characteristic to the student network. Instead of adopting the classic student-teacher learning of forcing the output of a student network to exactly mimic the soft targets produced by the teacher networks, we introduce two discriminators as in generative adversarial network (GAN) to transfer knowledge from two teachers to the student. The usage of discriminators relaxes the rigid coupling between the student and teachers. Extensive experiments on two benchmark datasets (i.e., CamRest and In-Car Assistant) demonstrate that TTOS significantly outperforms baseline methods. For reproducibility, we release the code and data at https://github.com/siat-nlp/TTOS.

1 Introduction

Task-oriented dialogue systems (TDSs), which help users to complete specific tasks with natural language, have attracted increasing attention recently due to the broad applications such as event scheduling and flight booking. Conventional TDSs have a complex pipeline (Williams and Young, 2007), which consists of modularly connected components for natural language understanding (NLU), dialogue state tracking (DST), and dialogue policy (DP). A limitation of 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. In addition, these methods usually require a large number of handcrafted features and labels, which may restrict the expressive power and learnability of the models.

To ameliorate the limitations with the conventional pipeline TDSs, great efforts have been made in designing deep neural network-based end-to-end solutions (Bordes et al., 2017; Eric et al., 2017; Madotto et al., 2018). Recent advances are overwhelmingly contributed by sequence-to-sequence (seq2seq) models (Bordes et al., 2017; Eric and Manning, 2017; Eric et al., 2017), which have taken the state-of-the-art of TDSs to a new level. These methods map dialogue context to output responses directly without explicitly providing handcrafted features and NLU/DST/DP labels, thus reduce human effort and are easily adapted to new domains.

Despite the effectiveness of previous studies, there are several technical challenges in building a TDS that is capable of retrieving accurate entries from the knowledge base (KB) and generating human-like responses. (1) Previous work (Carbonell, 1983) shows that users of TDSs tend to use succinct language which often omits entities or concepts made in previous utterances. However, seq2seq models often ignore how the conversation evolves as information progresses (Raghu et al., 2019) and thus result in generating incoherent and ungrammatical responses that are dominated by words appearing with high frequency in

* Min Yang is corresponding author

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 3498–3507,
November 16–20, 2020.

the training data. (2) Seq2seq models suffer from effectively reasoning over and incorporating KB information (Madotto et al., 2018). It is difficult to encode and decode the knowledge from a large and dynamic KB, making the response generation unstable. In addition, typically, a shared memory is used for both dialogue context and KB triples, making the TDSs struggle to reason over the two forms of data. Although some previous methods (Reddy et al., 2019) leverage separate memories for modeling dialogue context and KB facts, they either focus on capturing the dialogue patterns or retrieving accurate KB entities, but not both. One possible solution to the aforementioned problems is to explicitly encourage the seq2seq model to learn dialogue patterns and model the exterior KB knowledge retrieval with separate guidance for each.

In this study, we propose a "Two-Teacher One-Student" learning framework (TTOS) for building a high-quality TDS (student), where a student network is encouraged to integrate the knowledge from two expert teacher networks. Concretely, a KB-oriented teacher network (T_{KB}) is trained via reinforcement learning with entity score as the reward, which specializes in retrieving accurate KB entities; a dialogue pattern-oriented teacher network (T_{DP}) is trained via reinforcement learning with BLEU as the reward, which is expected to learn the language patterns of the dialogue (Eric and Manning, 2017), and thus specializes in generating coherent and grammatical responses. Afterwards, we optimize the student with distilled expert knowledge from two teacher networks. Our motivation is that the two teachers can provide different supervisory information that can be fully utilized through collaborative training. Instead of adopting the classic student-teacher learning strategy of forcing the output of a student network to exactly mimic the soft targets produced by the teacher networks, we employ the generative adversarial network (GAN) to transfer knowledge from two teachers to the student. To be more specific, the generator is the student network to produce dialogue responses, and the two discriminators distinguish the learned output representations from the student and teacher networks. By employing the output of the two discriminators as feedback, the student network can achieve collective success in both retrieving accurate KB entities and generating natural responses.

This paper has three main contributions listed as follows.

- We introduce a "Two-Teacher One-Student" learning framework for TDSs, where the student network benefits from the two teacher networks' complementary targets and achieves collective success in both retrieving accurate KB entities and generating natural responses.
- The expert knowledge is transferred from two teacher networks to the student network through two discriminators in our GAN-based approach. The usage of discriminators relaxes the rigid coupling between the student and teachers.
- Experimental results on In-Car Assistant and CamRest datasets demonstrate that TTOS achieves impressive results compared to the baseline methods across multiple evaluation metrics.

2 Related Work

2.1 Task-oriented Dialogue Systems

Task-oriented dialogue systems (TDSs), different from open-domain dialogue systems, are required to help users complete specific tasks with natural language. Conventional TDSs usually require a large number of handcrafted features, which may restrict the expressive power and learnability of the models (Williams and Young, 2007; Young et al., 2013). Inspired by the success of the sequence-to-sequence (seq2seq) models in text generation, there are several studies that build TDSs with the seq2seq model in an end-to-end trainable way. These methods have shown promising results recently since they have a great ability to learn the latent representations of dialogue context and are easily adapted to a new domain (Lei et al., 2018; Eric et al., 2017; Madotto et al., 2018).

However, as revealed by previous studies (Eric et al., 2017; Madotto et al., 2018), the performance of the seq2seq model deteriorates quickly with the increase of the length of the generated sequence. Therefore, how to improve the stability of the neural network models has gained increasing attention. (Eric et al., 2017) proposed a copy augmented seq2seq model by copying relevant information directly from the KB information. Mem2Seq (Madotto et al., 2018) and GLMP (Wu et al., 2019) further augmented memory-based

methods by incorporating copy mechanism (Gulcehre et al., 2016) to enable copying words from past dialog utterances or from KB when generating responses. Recently, separating memories for modeling dialog context and KB results are explored to improve the performance of TDSs (Raghu et al., 2019; Reddy et al., 2019; Chen et al., 2019). BossNet (Raghu et al., 2019) implicitly disentangled the language model from knowledge incorporation and thus enhanced the ability to copy unknown KB entries. Multi-level memory model (Reddy et al., 2019) represented the KB results using a multi-level memory instead of the form of triples. WMM2Seq (Chen et al., 2019) further employed a working memory to interact with two separated memories. Nevertheless, existing methods either achieve a good language model for the response generation or effective progress towards the KB modeling, but not both.

2.2 Student-teacher Learning Paradigm

In parallel, student-teacher learning has received intensive attention because of its excellent performance on various tasks. A typical application is to transfer knowledge from a large, powerful teacher network to a compact yet accurate student network, so as to boost the training process and the resulting performance (Watanabe et al., 2017; Bucilu and Niculescu-Mizil, 2006; Wang et al., 2018). For example, in (Bucilu and Niculescu-Mizil, 2006), a student network was encouraged to mimic the output of a teacher network via mean squared error. (You et al., 2017) proposed a dark knowledge distillation method, in which the student network accommodated the true labels and captured the structures among the labels. Instead of considering one single teacher network, several studies trained a student network by incorporating multiple teacher networks in the output layer or the hidden layers (Park and Kwak, 2019; You et al., 2017).

3 Our Methodology

Given the dialogue context $x = \{x_1, x_2, \ldots, x_M\}$ with M words and the system response $y = \{y_1, y_2, \ldots, y_T\}$ with T words, the dialogue system aims to optimize the generation probability of y conditioned on x, i.e., $p(y|x)$.

As illustrated in Figure 1, TTOS consists of three networks: a KB-oriented teacher network (T_{KB}) that is specialized for retrieving entities from KB, a dialogue pattern-oriented teacher network (T_{DP}) that is specialized for learning the dialogue patterns, and a student network (S) that tries to extract accurate KB entities and generate human-like responses. The three networks share the same network structure but different training strategies.

The learning procedure of TTOS contains two stages of training. In the first stage, the three networks are pre-trained independently with different training strategies. In particular, the student network is trained with supervised learning, while the two teacher networks T_{KB} and T_{DP} are trained via the reinforcement learning (RL) with goal-specific rewards (i.e., entity score and BLEU respectively), which can be viewed as experts towards the goals. Then, we employ GAN to learn the student network, where the generator is the student network to produce dialogue responses, and two discriminators distinguish the learned output responses from student and teacher networks. Next, we will introduce the three networks and the GAN-based student-teacher learning paradigm in detail.

3.1 Student Network

The student network a task-oriented dialogue system, which is responsible for both inquiring KB and generating human-like responses. In this study, the sequence-to-sequence (seq2seq) model (Luong et al., 2015) is used as the backbone to implement the student network. The seq2seq model additionally consists of a dialogue memory module and a KB memory module to store the information from the dialogue context and the retrieved KB entities, respectively.

Encoder Each input token in the dialogue context is converted to a fixed-length vector via an embedding layer. The input embedding sequence then goes into a layer of the bidirectional gated recurrent unit (BiGRU) (Chung et al., 2014) to learn the contextualized representation of the dialogue context, which is then passed into the dialogue memory.

Dialogue Memory and KB memory Different from Mem2Seq (Madotto et al., 2018), our dialogue memory is implemented with a dynamic key-value memory network (Zhang et al., 2017), which maintains a timely updated key memory to keep track of attention history and a fixed value-memory to store the dialogue context features throughout the whole generation process. In this way, the task-oriented dialogue system can keep track of the attention history along with the update-chain of

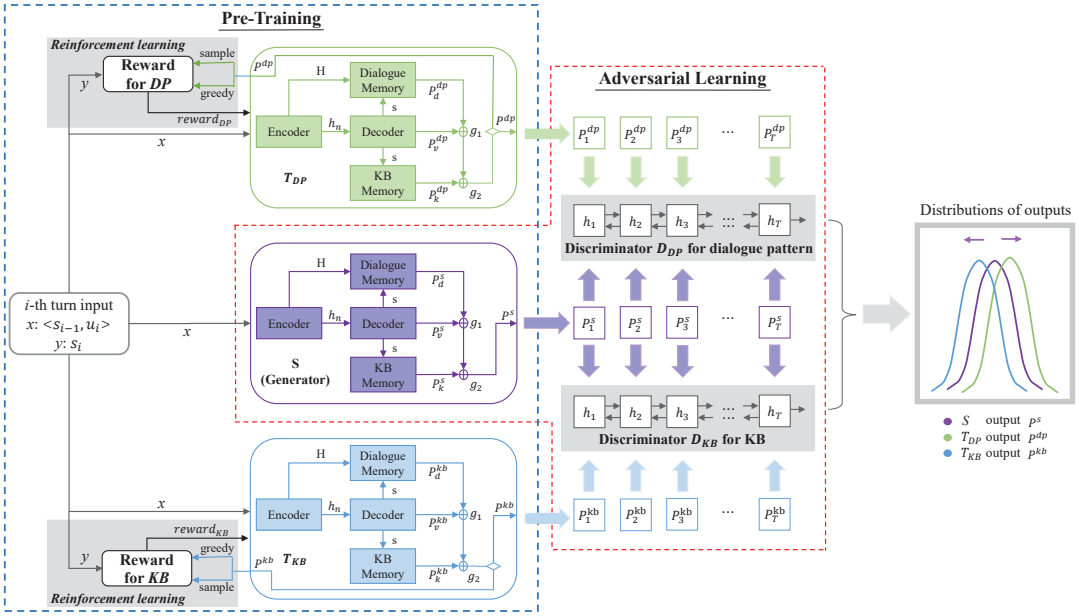

Figure 1: The overview of TTOS, which consists of two teacher networks (T_{KB} and T_{DP}) and a student network (S).

the decoder state, and therefore generate coherent and natural responses. In addition, we employ a separate KB memory, which is implemented with end-to-end memory networks (Sukhbaatar et al., 2015), to store the KB tuples.

Decoder The seq2seq model generates the response word by word. At decoding step t, the target word is either generated from the vocabulary or copied from the dialogue memory or KB memory. Formally, we use $P^s_{t,v}$, $P^s_{t,d}$, $P^s_{t,k}$ to denote the probabilities of generating the t-th target word from vocabulary, copying it from dialogue memory and KB memory, respectively. A soft gate g_1 controls whether a word is generated from vocabulary or copied from memories, and another gate g_2 determines which of the two memories is used to copy values. The final output distribution P_t for the t-th target word is calculated as:

$$P^s_t = g_1 P^s_{t,v} + (1 - g_1)\left[g_2 P^s_{t,d} + (1 - g_2) P^s_{t,k}\right] \quad (1)$$

The student network is optimized with supervised learning. We compute the loss function L_S of the student network as the cross-entropy between the output distribution P^s_t and the ground-truth target word y_t:

$$L_S = -\sum_{t=1}^{T} y_t \log(P^s_t) \quad (2)$$

where T is the length of the output response.

3.2 Two Teacher Networks

The two teacher networks share the same seq2seq network structure with the student network. Different from the student network that is trained with supervised learning, the two teacher networks T_{DP} and T_{KB} are trained via the reinforcement learning with goal-specific rewards (i.e., entity score and BLEU respectively), which can be viewed as experts towards different goals.

Dialogue Pattern-oriented Teacher Network The teacher network T_{DP} is specialized for learning the dialogue patterns so as to generate natural responses. To this end, we adopt the reinforcement learning technique, i.e., self-critical sequence training (SCST) algorithm (Rennie et al., 2017), to train the teacher network T_{DP} by using the BLEU as the reward function. As discussed in (Eric and Manning, 2017), BLEU can be used to gauge the model's ability to accurately generate the dialogue patterns seen in the training data. In particular, we generate two separate output sequences at each training iteration: (1) the output y^s that is obtained by sampling from the output distribution P^{dp}_t, and (2) the baseline output $\hat{y}$ that is obtained by maximizing the output distribution with a greedy search.

Following (Rennie et al., 2017), the loss function

of the SCST algorithm for T_{DP} can be derived as:

$$L_{DP} = -(\text{BLEU}(y^s) - \text{BLEU}(\hat{y})) \sum_{t=1}^{T} \log(P_t^{dp}) \tag{3}$$

KB-oriented Teacher Network The teacher network T_{KB} is specialized for retrieving accurate KB entities from KB to accomplish the task. We employ SCST to optimize the network T_{KB} by using entity F1 score (Ent.F1) as the reward function. The entity F1 metric evaluates the model's ability to generate relevant entities from the underlying KB. Similar to the network T_{DP}, the loss function of the SCST algorithm for T_{KB} can be derived as:

$$L_{KB} = -(\text{Ent.F1}(y^s) - \text{Ent.F1}(\hat{y})) \sum_{t=1}^{T} \log(P_t^{kb}) \tag{4}$$

where P_t^{kb} is the output distribution of the teacher network T_{KB}.

3.3 Improving Student Network with GAN

After pre-training the three networks, we further train the student network to amalgamate expert knowledge from the two teacher networks. Different from previous student-teacher learning methods (Hinton et al., 2015; Kim and Rush, 2016) which force the output of the student network to exactly mimic the soft targets produced by the teacher networks, we introduce two discriminators as GAN to transfer knowledge from the two teacher networks to the student network. The two discriminators are trained to distinguish the learned output representations from student and teacher networks, while the student network (generator) is adversarially trained to produce dialogue responses to fool the discriminators. To be more specific, a discriminator D_{DP}, a binary classifier implemented with a BiGRU, is proposed to distinguish the output distributions generated by the student S and the teacher T_{DP}. Similarly, another binary classifier discriminator D_{KB} is employed to distinguish whether the output distribution is from the student S or the teacher T_{KB}.

By alternatively updating the student and the two discriminators in an adversarial process, the expert knowledge transferred from discriminators can eventually guide the student to produce responses similar to the responses generated by the two teachers. The details of the adversarial training are summarized in Algorithm 1.

3.3.1 Discriminator Update

The two discriminators and the student (generator) are alternatively updated in the GAN-based approach. We first introduce the update process of the discriminators. The discriminators are two binary classifiers that are trained to distinguish the output responses generated by the student and teachers. For each discriminator, we encode the output distribution P^o with a BiGRU as it shows great effectiveness in text classification. The last hidden state ($\mathbf{h}_T$) is then passed to an output layer (sigmoid) whose output is the probability of being "true". Formally, given the output distribution P_t^o at t-th time step, the binary classifier (discriminator) D is defined as:

$$\mathbf{h}_t = \text{BiGRU}(P_t^o, \mathbf{h}_{t-1}), \quad t \in [1, T] \tag{5}$$

$$D(P^o) = \text{sigmoid}(W\mathbf{h}_T) \tag{6}$$

where W is a learnable parameter, $\mathbf{h}_t$ is the hidden state at the t-th time step. In this way, we can obtain the dialogue pattern-oriented discriminator D_{DP} that predicts whether the input sequence is generated by the teacher T_{DP} and the KB-oriented discriminator D_{KB} that predicts whether the input sequence is generated by the teacher T_{KB}.

When training the discriminators, we fix the student (generator). The two discriminators (i.e., L_{DP} and L_{KB}) are trained to minimize the probability of assigning the incorrect labels to the output distributions of the student and teacher networks:

$$L_{DP} = -\log(1 - D_{DP}(P^s)) - \log(D_{DP}(P^{dp})) \tag{7}$$

$$L_{KB} = -\log(1 - D_{KB}(P^s)) - \log(D_{KB}(P^{kb})) \tag{8}$$

$$L_D = L_{DP} + L_{KB} \tag{9}$$

where are P^s, P^{dp}, P^{kb} are the output distributions produced by S, T_{DP}, T_{KB}, respectively. L_D is the final objective function for the discriminator update.

3.3.2 Student Update

In each iteration, we update the student network (generator) S after updating the two discriminator networks. When updating the student network S, we try to fool the two discriminators and minimize the adversarial loss L_G which is defined as:

$$L_G = -\log(D_{DP}(P^s)) - \log(D_{KB}(P^s)) \tag{10}$$

Algorithm 1 Adversarial training procedure.

Input: Three pre-trained networks S, T_{DP}, T_{KB}, and training dataset (X, Y).
Output: Student network S that integrates expert knowledge from two teachers.

1: Initialize the generator ($G = S$) and two discriminators (D_{DP} and D_{KB}) in GAN;
2: **repeat**
3: Sample an instance (x, y) from the training data;
4: Produce the output distributions P^{dp} and P^{kb} by two teachers T_{DP} and T_{KB};
5: Produce output distribution P^s by student S;
6: Fix generator G and update discriminators D_{DP} and D_{KB} by minimizing Eq. (9) via gradient descent.
7: Fix the discriminators and update generator G by minimizing Eq. (11) via gradient descent.
8: **until** convergence

The final loss function $\tilde{L}_S$ for the student network S is computed as:

$$\tilde{L}_S = L_S + \alpha L_G \tag{11}$$

where α is a scalar that determines the importance of the adversarial loss L_G of the student network.

4 Experimental Setup

Dataset We evaluate the proposed TTOS model on two widely used multi-turn task-oriented dialogue datasets: CamRest (Wen et al., 2016) and In-Car Assistant (Eric and Manning, 2017). The CamRest dataset is composed of 676 human-to-human multi-turn conversations in the restaurant reservation domain. The average number of turns per dialogue is about 5. Following in (Reddy et al., 2019), we divide the dataset into training/validation/testing sets with 406/135/135 dialogues respectively. The In-Car Assistant dataset contains 3031 multi-turn dialogues, which are divided into 2425/302/304 dialogues for training/validation/testing, respectively. In-Car Assistant includes three distinct domains: calendar scheduling, weather information retrieval, and point-of-interest navigation. There are 2.6 turns on average per dialogue. Compared to CamRest, the In-Car Assistant dataset is more diverse in the utterances, and the KB information is also more complicated.

Training Details The grid search algorithm (Bergstra et al., 2013) is applied on the validation set to automatically tune the hyper-parameters. We use the 300-dimensional word2vec vectors (Mikolov et al., 2013) to initialize the word embeddings. The size of the GRU hidden units is set to 256. The number of hops for the memory network is set to 3. The recurrent weight parameters are initialized as orthogonal matrices. We initialize the other weight parameters with the normal distribution $\mathcal{N}(0, 0.01)$ and set the bias terms to zero. To stabilize the process of training GAN, we use Adam optimizer (Kingma and Ba, 2014) with a relatively small initial learning rate of $1e^{-4}$ to train the model. The batch size is set to 8. The step ratio of G and D is set to 1:1 for reaching a training balance. We set the value of α to 1.0, because a too large α value will make student rely excessively on teachers' outputs without concrete guidance at each time step, while a loss with too low α value cannot guide the student to fool the two discriminators, which may make the adversarial training process unstable. In addition, we also apply dropout (dropout rate=0.2) on several layers of generator, but not for discriminators.

It is noteworthy that we first pre-train the three networks separately by optimizing the networks with different training strategies. Then, we switch to the GAN training to learn the student network by amalgamating knowledge from the two teacher networks.

Model	BLEU	Entity F1
Seq2Seq	7.9	17.6
Seq2Seq+Attn	7.7	21.4
Ptr-Unk	5.1	16.4
Mem2Seq	13.51	33.57
BossNet	15.20	43.10
ECET	18.50	58.60
GLMP	16.70	50.61
TTOS (Ours)	**20.45**	**61.50**
$S^{\#}$	18.77	58.80
T_DP	20.49	57.03
T_KB	18.77	59.26

Table 1: Automatic evaluation results on CamRest dataset. $S^{\#}$, T_DP, and T_KB denote the pre-trained student and teacher networks before adversarial training.

Compared Methods We compare TTOS with several state-of-the-art task-oriented dialogue systems, including Seq2Seq/+Attn (Luong et al., 2015), Seq2Seq model with copy mechanism (Ptr-Unk) (Gulcehre et al., 2016), network-based seq2seq (Mem2Seq) (Madotto et al., 2018), bag-of-sequences memory network (BossNet) (Raghu et al., 2019), entity-consistent network with KB retriever (ECET) (Qin et al., 2019), global-to-

Model	BLEU	Ent. F1	Sch.F1	Wea.F1	Nav.F1
Seq2Seq	8.4	10.3	9.7	14.1	7.0
Seq2Seq+Attn	9.3	19.9	23.4	25.6	10.8
Ptr-Unk	8.3	22.7	26.9	26.7	14.9
Mem2Seq	12.6	33.4	49.3	32.8	20.0
BossNet	8.3	35.9	50.2	34.5	21.6
ECET	14.1	53.7	54.5	52.2	**55.6**
GLMP	14.79	**59.97**	**69.56**	62.58	52.98
TTOS (Ours)	**17.35**	55.38	63.50	**64.09**	45.90
$S^{\#}$	16.80	51.84	60.71	62.67	40.76
T_DP	17.23	51.49	61.18	63.78	39.14
T_KB	17.05	55.88	67.53	63.71	44.86

Table 2: Automatic evaluation results on In-Car Assistant dataset.

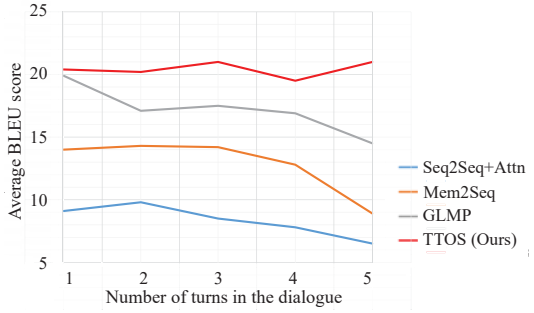

Figure 2: The performance of TTOS and baselines on CamRest dataset with the increase of dialogue turns.

local memory pointer network (GLMP) (Wu et al., 2019).

Automatic Evaluation Metrics Following previous works (Madotto et al., 2018; Wu et al., 2019), we evaluate TTOS and compared methods on two automatic evaluation metrics: BLEU (Papineni et al., 2002) and entity F1 (Madotto et al., 2018) scores. BLEU calculates n-gram overlaps between the generated response and the gold response, which could gauge the model's ability to accurately generate the dialogue patterns seen in our data. BLEU shows a comparatively strong correlation with the human assessment on task-oriented systems (Sharma et al., 2017). Entity F1 is computed by micro-averaging the precision and recall over KB entities in the entire set of system responses, which evaluates the ability of the TDSs to generate relevant entities to accomplish specific tasks by inquiring the provided KBs.

5 Experimental Results

Automatic Evaluation Results For each test instance, we use the response generated by the student network (learned by adversarial training) as the final output response. Table 1 shows the automatic evaluation results of TTOS and baseline methods. From the results, we can observe that TTOS achieves substantially and significantly better performance than the compared methods over the two evaluation metrics. Mem2Seq and BossNet consistently perform better than Seq2Seq(+Attn) and Ptr-Unk. This verifies the effectiveness of memory networks in incorporating KB information into the seq2seq model for generating better responses. GLMP has achieved a strong improvement over both BLEU and entity F1 scores over the previous models, which is mainly benefited from its global and local memory pointers to guide the KB attention and response generation. TTOS performs even better than GLMP on both metrics. Furthermore, Figure 2 shows the changes in average BLEU scores of TTOS and several baselines along with the increase of dialog turns on the CamRest dataset. The BLEU scores of the baseline models decrease sharply after four turns while TTOS achieves much more stable performance even in the last few turns. This verifies the effectiveness and stableness of TTOS in retrieving accurate KB entities and generating human-like responses. We observe similar trends on the In-Car Assistant dataset, as shown in Table 2. TTOS achieves competitive results on both BLEU and entity F1 compared to most of the baseline methods. In particular, TTOS achieves the best BLEU on the whole dataset and best entity F1 score in the domain of weather information retrieval.

Note that most methods (e.g., TTOS, GLMP, BossNet, Mem2Seq) except ECET achieve much worse entity F1 on Nav. sub-domain than on Sch./Wea. sub-domains. This may be because the KB of Nav. sub-domain is more complicated than the KBs of other two sub-domains. In particular, each entry in the KB of Nav. sub-domain has more complex attributes. On the other hand, the dialogues in Nav. sub-domain often involve multi-step reasoning, while single-step reasoning is usually involved in Sch./Wea. sub-domains. For example, for the request "give me directions to the nearest grocery_store", the dialogue system should first restrict the location type to be "grocery_store" and then find the nearest location based on distance attribute. In the future, we plan to design more advanced techniques to integrate the KB knowledge and further improve the multi-step reasoning ability of TTOS.

Restaurant	Area	Address	Food	Price	Phone	Postcode
pizza_hut_cherry_hinton	south	g4_cambridge_leisure_park_clifton_way_cherry_hinton	italian	moderate	01223_323737	cb17dy
frankie_and_bennys	south	cambridge_leisure_park_clifton_way_cherry_hinton	italian	expensive	01223_412430	cb17dy
...	...	...	...	...	...	...

Role	Turn	Utterance
User	1	i want a restaurant in the south part of town and serves panasian food.
System	1	sorry, there are no restaurants in the **south** part of town serving panasian food.
Mem2Seq	1	pizza_hut_cherry_hinton is a restaurant in the **south** part of town.
GLMP	1	pizza_hut_cherry_hinton serves panasian food in the **south** part of town.
T_DP	1	there are no restaurants serving panasian food in the **south** part of town.
T_KB	1	there are no **south** restaurants serving panasian food in the v part of town.
$S^{\#}$	1	there are no restaurants in the **south** part of town.
TTOS ($S^{\#}$)	1	there are no **south** restaurants serving panasian food in the **south** part of town.
User	2	italian food please. phone number and postcode please.
System	2	**frankie_and_bennys** serves **italian** food in the **south** part of town. the phone number is **01223_412430**, and the postcode is **cb17dy**.
Mem2Seq	2	their phone number is is 01223_323737 and their postcode is **cb17dy**.
GLMP	2	their phone number is **01223_412430** and their postcode is **cb17dy** in the **south** part of town.
T_DP	2	**frankie_and_bennys** serves **italian** food. their phone number is 01223_323737 and their postcode is **cb17dy** in the **south** part of town.
T_KB	2	**frankie_and_bennys frankie_and_bennys frankie_and_bennys** is **01223_412430** and **cb17dy** in the **south** part of town.
$S^{\#}$	2	**frankie_and_bennys** their phone number is 01223_323737 and their postcode is **cb17dy**.
TTOS ($S^{\#}$)	2	**frankie_and_bennys** serves **italian** food. their phone number is **01223_412430** and their postcode is **cb17dy**.
User	3	sorry what type of food do they serve?
System	3	they serve **italian** food.
Mem2Seq	3	pizza_hut_cherry_hinton serve serve food in the south part of town and in the south part of town.
GLMP	3	pizza_hut_cherry_hinton serve **italian** food in the south part of town.
T_DP	3	they serves **italian** food in the south part of town.
T_KB	3	frankie_and_bennys serves **italian** food.
$S^{\#}$	3	they serves **italian** food.
TTOS ($S^{\#}$)	3	frankie_and_bennys serves **italian** food.
User	4	what is the price range?
System	4	they are quite **expensive**.
Mem2Seq	4	pizza_hut_cherry_hinton is moderate restaurant in the south part of town and in the south part of town.
GLMP	4	they serve moderate food in the south part of town.
T_DP	4	they are **expensive** in the south part of town.
T_KB	4	frankie_and_bennys are **expensive**.
$S^{\#}$	4	they serves moderate food.
TTOS ($S^{\#}$)	4	frankie_and_bennys serves **expensive** food.

Table 3: Case study of generated responses by TTOS and baselines on CamRest test set. Gold entities in responses are highlighted in bold.

Model	CamRest		In-Car	
	Infor.	Humanlike.	Infor.	Humanlike.
Mem2Seq	3.33	3.78	3.76	3.65
BossNet	3.74	4.11	3.78	3.73
GLMP	4.05	4.15	4.21	4.09
TTOS (Ours)	**4.19**	**4.28**	**4.22**	**4.18**

Table 4: Human evaluation results on CamRest and In-Car Assistant.

Model	In-Car		CamRest	
	BLEU	Ent. F1	BLEU	Ent.F1
w/o TDP	17.02	54.69	19.56	60.66
w/o TKB	17.11	52.96	19.98	59.88
w/o GAN	16.80	51.84	18.77	58.80
TTOS with KD	16.95	53.74	19.79	60.10
TTOS with GAN	**17.35**	**55.38**	**20.45**	**61.50**

Table 5: Ablation results of TTOS on two datasets.

Case Study To evaluate the proposed model qualitatively, we choose an exemplary dialogue from the CamRest test set and illustrate some generated responses by TTOS and the compared methods in Table 3. We observe that Mem2Seq fails to understand the dialogue context and thus generates irrelevant responses. GLMP generates more readable responses than Mem2Seq but fails to extract correct KB entities. In particular, the performance of GLMP deteriorates significantly with the increase of dialogue turns. Compared to GLMP, TTOS can retrieve more accurate KB entities and generate more natural responses, especially in the last few turns. This verifies that TTOS can identify key entities and keep track of dialog context from previous turns.

We also provide the generated responses by two teachers (T_DP and T_KB) and the pre-trained student ($S^{\#}$), to analyze where the empirical gains come from. From Table 3, we can observe that T_DP can captures the dialogue pattern while the response generated by $S^{\#}$ fails to extract "serving panasian food" that modifies the word "restaurants". On the other hand, the teacher T_KB works well for entity retrieval. For example, in the fourth turn, T_KB can trigger the accurate entity word "expensive", which is not recognized by $S^{\#}$.

Human Evaluation Results Similar to the previous work (Wu et al., 2019), we use human evaluation to evaluate the generated responses from two perspectives: informativeness (*Infor.*) and human-likeness (*Humanlike.*). Specifically, we randomly select 100 dialogues from the CamRest and In-Car Assistant test sets, and invite three annotators to

independently assign two scores (i.e., informativeness and human-likeness scores) from 1 to 5 for each generated response. A higher score means better performance. The agreement ratios computed with Fleiss' kappa (Fleiss, 1971) are 0.58 on CamRest and 0.51 on In-Car Assistant, showing moderate agreement. We report the average rating scores from all annotators as the final human evaluation results. As shown in Table 4, TTOS outperforms the compared methods on both informativeness and human-likeness by a noticeable margin, which is consistent with the automatic evaluation.

Ablation Study To investigate the effectiveness of each module in TTOS framework, we conduct ablation test in terms of removing the teacher T_DP (w/o TDP), removing the teacher T_KB (w/o TKB), removing the GAN-based student-teacher learning (w/o GAN). In addition, we also replace the GAN-based student-teacher learning with the standard knowledge distillation method (denoted as TTOS with KD). The experimental results are reported in 5. The performance of TTOS drops sharply when we discard the two teachers and the GAN-based student-teacher learning. This is within our expectation since TTOS achieves collective success from two teachers that are specialized for two different goals through two discriminators in GAN-based approach.

6 Conclusion

In this paper, we propose a novel "Two-Teacher One-Student" learning framework (TTOS) for task-oriented dialogue, which aims to improve the performance of the task-oriented dialogue system in retrieving accurate entries from KB and generating human-like responses simultaneously. With adversarial learning, we train the student network to amalgamate expert knowledge naturally from the two teacher networks for the above two goals. The experimental results on two benchmark datasets demonstrated that our model achieves impressive results compared to the state-of-the-art task-oriented dialogue systems.

Acknowledgement

This work was partially supported by National Natural Science Foundation of China (No. 61906185), the Natural Science Foundation of Guangdong Province of China (No. 2019A1515011705, 2018A030313943), Shenzhen Basic Research Foundation (No. JCYJ20180302145607677, JCYJ20190808182805919), the Youth Innovation Promotion Association of CAS.

References

James Bergstra, Daniel Yamins, and David Cox. 2013. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pages 115–123.

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2017. Learning End-to-end Goal-oriented Dialog. *ICLR*.

Rich Bucilu, Cristian and Caruana and Alexandru Niculescu-Mizil. 2006. Model compression. In *SIGKDD*, pages 535–541.

Jaime G Carbonell. 1983. Discourse pragmatics and ellipsis resolution in task-oriented natural language interfaces. In *ACL*, pages 164–168.

Xiuyi Chen, Jiaming Xu, and Bo Xu. 2019. A working memory model for task-oriented dialog response generation. In *ACL*, pages 2687–2693.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Mihail Eric, Lakshmi Krishnan, Francois Charette, and Christopher D Manning. 2017. Key-value retrieval networks for task-oriented dialogue. In *SIGDIAL*, pages 37–49.

Mihail Eric and Christopher D Manning. 2017. A Copy-augmented Sequence-to-sequence Architecture Gives Good Performance on Task-oriented Dialogue. *EACL*, page 468.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *ACL*, pages 140–149.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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 *ACL*, pages 1437–1447.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421.

Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2018. Mem2seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems. In *ACL*, pages 1468–1478.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318.

SeongUk Park and Nojun Kwak. 2019. Feed: Feature-level ensemble for knowledge distillation. *arXiv preprint arXiv:1909.10754*.

Libo Qin, Yijia Liu, Wanxiang Che, Haoyang Wen, Yangming Li, and Ting Liu. 2019. Entity-consistent end-to-end task-oriented dialogue system with kb retriever. In *EMNLP/IJCNLP*.

Dinesh Raghu, Nikhil Gupta, et al. 2019. Disentangling language and knowledge in task-oriented dialogs. In *NAACL-HLT*, pages 1239–1255.

Revanth Reddy, Danish Contractor, Dinesh Raghu, and Sachindra Joshi. 2019. Multi-level memory for task oriented dialogs. In *NAACL-HLT*, pages 3744–3754.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *CVPR*, pages 7008–7024.

Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. 2017. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *ArXiv*, abs/1706.09799.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *NIPS*, pages 2440–2448.

Yunhe Wang, Chang Xu, Chao Xu, and Dacheng Tao. 2018. Adversarial learning of portable student networks. In *AAAI*.

Shinji Watanabe, Takaaki Hori, Jonathan Le Roux, and John R Hershey. 2017. Student-teacher network learning with enhanced features. In *ICASSP*, pages 5275–5279.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Lina M Rojas Barahona, Pei-Hao Su, Stefan Ultes, David Vandyke, and Steve Young. 2016. Conditional generation and snapshot learning in neural dialogue systems. In *EMNLP*, pages 2153–2162.

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, Richard Socher, and Caiming Xiong. 2019. Global-to-local memory pointer networks for task-oriented dialogue. In *ICLR*.

Shan You, Chang Xu, Chao Xu, and Dacheng Tao. 2017. Learning from multiple teacher networks. In *SIGKDD*, pages 1285–1294. ACM.

Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. 2013. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179.

Jiani Zhang, Xingjian Shi, Irwin King, and Dityan Yeung. 2017. Dynamic key-value memory networks for knowledge tracing. *The Web Conference*, pages 765–774.