Exploring Auxiliary Reasoning Tasks for Task-oriented Dialog Systems with Meta Cooperative Learning

Bowen Qin[1,2], Min Yang[1*], Lidong Bing, Qingshan Jiang[1], Chengming Li[1], Ruifeng Xu[3]

[1]Shenzhen Key Laboratory for High Performance Data Mining,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
[3]Harbin Institute of Technology (Shenzhen)
{bw.qin, min.yang, qs.jiang, cm.li}@siat.ac.cn, binglidong@gmail.com, xuruifeng@hit.edu.cn

Abstract

In this paper, we propose a Meta Cooperative Learning (MCL) framework for task-oriented dialog systems (TDSs). Our model consists of an auxiliary KB reasoning task for learning meta KB knowledge, an auxiliary dialogue reasoning task for learning dialogue patterns, and a TDS task (primary task) that aims at not only retrieving accurate entities from KB but also generating natural responses, which are coordinated to achieve collective success in both retrieving accurate KB entities and generating human-like responses via meta learning. Concretely, the dialog generation model amalgamates complementary meta KB and dialog knowledge from two novel auxiliary reasoning tasks that together provide integrated guidance to build a high-quality TDS by adding regularization terms to force primary network to produce similar results to auxiliary networks. While MCL automatically learns appropriate labels for the two auxiliary reasoning tasks from the primary task, without requiring access to any further data. The key idea behind MCL is to use the performance of the primary task, which is trained alongside the auxiliary tasks in one iteration, to improve the auxiliary labels for the next iteration with meta learning. Experimental results on three benchmark datasets show that MCL can generate higher quality responses compared to several strong baselines in terms of both automatic and human evaluations. Code to reproduce the results in this paper is available at: https://github.com/siat-nlp/MCL.

1 Introduction

Task-oriented dialog systems (TDSs), which aim at facilitating users to accomplish certain tasks with natural language, have raised increasing interest due to their broad applicability, such as weather inquiry and restaurant reservation. Different from open-domain dialog systems which perform chit-chat, the ability to query external KB is essential in TDSs since the responses are guided not only by dialog context but also by the query results (see Table 1). Conventional pipeline TDSs decompose the dialog generation into separate modularized pipelines, which rely on handcrafted labels, features and templates. However, pipelining dialog system components often leads to error propagation. In addition, these methods often require a large number of handcrafted features and labels, which may create a substantial barrier for adapting the learned TDSs to new domains.

Address	Distance	POI type	POI
434_arastradero_rd	6_miles	shopping_center	ravenswood_shopping_center
53_university_av	8_miles	gas_station	shell
481_amaranta_ave	2_miles	parking_garage	palo_alto_garage_r
753_university_ave	3_miles	rest_stop	comfort_inn

Role	Turn	Utterance
User	1	where can i get gas?
System (Gold)	1	the nearest gas_station is 8_miles away.
Seq2seq	1	I have a away from would you like the address.
MLM	1	It is 8_miles away.
MCL	1	There is a gas_station, Shell is 8_miles away.
User	1	what is the address?
System (Gold)	2	shell is at 53_university_av.
Seq2seq	1	The is at at.
MLM	1	shopping_center is located at 434_arasadero_rd. shell is located at 53_university_av.
MCL	1	shell is located at 53_university_av, i sent its position to your screen.

Table 1: A dialog along with the KB information from the CamRest dataset for restaurant reservation.

Recently, great efforts have been made in building end-to-end TDSs with the sequence-to-sequence model (Seq2Seq) (Bordes, Boureau, and Weston 2017; Eric and Manning 2017; Eric et al. 2017), which bring the results of good flexibility and quality. The primary idea behind the Seq2Seq model is to leverage an encoder to create a vector representation of dialog history and KB information, and then feed the learned representation into a decoder to produce a response word by word. For example, GLMP (Wu, Socher, and Xiong 2019) is a representative end-to-end TDS, which incorporates KB information into the Seq2Seq model by using a global memory pointer to filter the KB for relevant information and instantiating the slots with a local memory pointer. The end-to-end TDSs map dialog contexts into output responses directly without acquiring explicit dialog state and dialog policy labels, thus reduce human effort and are easily adapted to new domains.

Despite the remarkable progress of previous works, there are still several challenges for extracting accurate entities from KB and generating natural responses. First, the Seq2Seq model struggles to effectively reason over the complex KB and integrate the KB entities into dialog generation, making it unstable to generate the dialog responses (Wu, Socher,

[*]Min Yang is corresponding author (min.yang@siat.ac.cn).

and Xiong 2019). Taking the dialog in Table 1 as an example, when answering the user question in the first turn, the Seq2Seq model is prone to generate unrelated response to the dialog history without effectively modelling the KB information. Second, previous study (Carbonell 1983) argues that the users of a dialog system are prone to utilize succinct language and often drop entities appeared in historical utterances. Nevertheless, the Seq2Seq model tends to neglect how the conversation evolves as information progresses and thus leads to incoherent and ungrammatical responses that are dominated by words appearing with high frequency in the training data (Wang et al. 2020). As shown in Table 1, it is difficult for the Seq2Seq model to infer the address of gas station from a long concatenated dialog context when answering the second user question.

One possible solution to the aforementioned challenges is to leverage auxiliary tasks to explicitly assist the dialog generation model in learning dialog patterns and reasoning over the large KB, respectively. The auxiliary labels results in additional relevant knowledge being available, which otherwise would not have been learned from training only on the task-oriented dialog generation task. The broader support of these knowledge, across new interpretations of input data, then boost the performance and generalisation of the primary task. In this way, the TDSs can integrate the expert knowledge from the auxiliary tasks and obtain comprehensive performance for dialog generation.

In this paper, we propose a Meta Cooperative Learning (MCL) framework for task-oriented dialog generation, in which the dialog generation model amalgamates implicit meta knowledge from two novel auxiliary reasoning tasks without requiring access to any further data. Specifically, our model consists of three networks: (i) an auxiliary KB reasoning task for learning meta KB knowledge; (ii) an auxiliary dialogue reasoning task for learning dialogue patterns; (iii) a TDS task (primary task) for not only retrieving accurate entities from KB but also generating natural responses. MCL coordinates the three models to obtain collective success in both extracting accurate KB entities and producing human-like responses. The primary network learns complementary meta KB and dialog knowledge from two novel auxiliary reasoning tasks by adding corresponding regularization terms to force primary network to produce similar results to auxiliary networks. While MCL automatically learns appropriate labels for the two auxiliary reasoning tasks from the primary task via meta learning, without requiring access to any further data. This is achieved by defining the objectives for two auxiliary networks as functions of the primary network's performance on the training data. The usage of "gradient by gradient" strategy makes the primary task adjust to the learning state of auxiliary tasks, and improves the auxiliary tasks accordingly. As shown in Table 1, MCL can retrieve accurate KB entities and produce fluent responses thanks to the help of two auxiliary tasks.

We summarize our main contributions as follows:

- We introduce auxiliary KB and dialogue reasoning tasks to learn implicit meta knowledge from the KB and dialog context respectively, which together provide comprehensive guidance to our task-oriented dialogue generation model in extracting accurate entities from KB and producing human-like responses simultaneously.
- We propose a meta cooperative learning method to automatically learn appropriate labels for the two auxiliary reasoning tasks from the primary task with the usage of "gradient by gradient" strategy, which removes the need for manual labelling of auxiliary tasks or any further data. In this way, the primary network with MCL outperforms the single-task learning, even though the three networks adopt the same amount of training data.
- Experiments on three benchmark datasets demonstrate that MCL significantly outperforms the strong baselines in terms of quantitative evaluation and human evaluation.

2 Related Work

2.1 Task-oriented Dialog Systems

Task-oriented dialog systems allow users to seek information and complete complex tasks using natural language in an interactive manner. Recently, end-to-end TDSs (Lei et al. 2018; Gulcehre et al. 2016) have gained increasing attention, since they directly map dialog history to target responses and consequently save human effort to annotate hand-crafted state labels. So far, the Seq2Seq models have dominated the study of TDSs (Eric et al. 2017; Lei et al. 2018), since they have the ability to learn latent representations for dialog history that are easily adapted to new domains. To effectively incorporate KB knowledge and perform KB reasoning, memory networks have been explored in TDSs (Madotto, Wu, and Fung 2018; Wu, Socher, and Xiong 2019; Chen, Xu, and Xu 2019). GLMP (Wu, Socher, and Xiong 2019) incorporated KB information into Seq2Seq model by using a global memory pointer to extract relevant KB information and instantiating the slots with a local memory pointer. Qin et al. (2020) proposed a dynamic fusion network to capture the correlation between domains in multi-domain TDSs.

There are also several works applying separate memories to model dialog history and KB information so as to further enhance the performance of TDSs (Raghu, Gupta et al. 2019; Reddy et al. 2019; Chen, Xu, and Xu 2019; Wang et al. 2020; He et al. 2021). For example, multi-level memory (Reddy et al. 2019) utilized a multi-level memory to model the KB results, rather than using the form of triples. WMM2Seq (Chen, Xu, and Xu 2019) introduced a working memory to coordinate two separated memories. DDMN (Wang et al. 2020) employed a dual dynamic memory network to model the dialog context and KB information. However, previous models do not explicitly explore the auxiliary reasoning tasks to help model the dialog context and KB information.

MCL is also closely related to TTOS (He et al. 2020) that employs adversarial learning to transfer knowledge from two teacher networks (trained with masked dialogs) to the TDS (student network). Different from TTOS (He et al. 2020), our MCL model learns to generate useful auxiliary labels for two auxiliary reasoning tasks automatically based on the previous learning episodes via meta learning, so as to assist the TDS (primary task) in extracting accurate KB entities and producing natural responses.

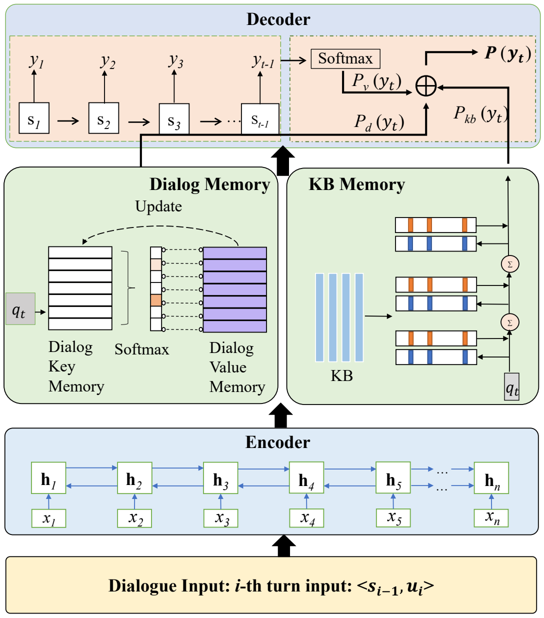

Figure 1: The architecture of the base model.

2.2 Meta Learning in NLP

Meta learning can be understood as learning to learn, which refers to the process of improving a learning algorithm over multiple learning episodes (Thrun and Pratt 1998; Hospedales et al. 2020). Meta-learning has recently attracted extensive attention in natural language processing (NLP), especially when addressing low resource issues in various NLP tasks. For example, Mi et al. (2019) applied model-agnostic meta-learning (MAML) (Finn, Abbeel, and Levine 2017) to solve natural language generation in task-oriented dialog systems in the low-resource scenario. Madotto et al. (2019) learned different personas as different tasks via meta-learning algorithms for personalized dialog systems. Yan et al. (2020) extended meta learning by incorporating multiple training sources for low-resource multi-choice question answering. However, the aforementioned studies directly use MAML algorithm to mitigate the low-resource problems in NLP applications, and auxiliary reasoning tasks are explored to cooperatively enhance the primary task.

3 Our Methodology

Given the input: (1) dialog history that includes a sequence of historical user utterances $\{u_1, \ldots, u_c\}$ and system response utterances $\{s_1, \ldots, s_{c-1}\}$, and (2) KB tuples $\{b_1, \ldots, b_l\}$, the goal of the task-oriented dialog generation (primary task) is to generate the next system response $s_c = \{y_1, y_2, \ldots, y_T\}$ word by word, where c and l represent the numbers of utterances and KB tuples, T is the length of the generated response. In addition, we also design a KB reasoning auxiliary task to extract entities from KB and a dialog reasoning auxiliary task to learn dialog patterns from the perspective of language modeling. Next, we elaborate on the primary task and two auxiliary reasoning tasks in detail.

3.1 Task-oriented Dialog Generation

The primary model is a task-oriented dialog system, which is responsible for both inquiring KB and producing natural responses. The backbone of our TDS is inspired by (Wang et al. 2020). As illustrated in Figure 1, our proposed task-oriented dialog generation model is composed of four primary components: a dialog encoder, a dialog memory, a KB memory, and a response decoder.

Dialog Encoder Given dialog history with a sequence of user utterances and system responses, our dialog encoder encodes dialog context turn by turn. In particular, the first turn input of the encoder is u_1. For the i-th ($i > 1$) turn, the input is $\{s_{i-1}, u_i\}$ consisting of the last system response s_{i-1} and current user request u_i. In this paper, we define the input $\{s_{i-1}, u_i\}$ at each turn as dialog context, which is denoted as a sequence of tokens $X = (x_1, x_2, \ldots, x_m)$, where m represents the sequence length. First, each token is converted into a word embedding through an embedding layer. Then, we apply a BiGRU (Chung et al. 2014) to encode the dialog context into hidden states:

$$\mathbf{h}_t = \text{BiGRU}(\mathbf{e}(x_t), \mathbf{h}_{t-1}) \quad (1)$$

where $\mathbf{e}(x_t)$ denotes the embedding of word x_t. The forward and backward hidden states are concatenated to form the output of the encoder, denoted as $\mathbf{H} = (\mathbf{h}_1, \ldots, \mathbf{h}_m)$, which is then passed into the dialog memory.

Dialog Memory We propose a dialog memory to reason over dialog context, which is implemented with a dynamic key-value memory network (Wang et al. 2020). The dialog memory network contains a dialog key memory and a dialog value memory. The two memories are initialized with the dialog hidden states of the first turn and maintained throughout the whole conversation. The dialog key memory keeps updated at each turn for tracking the dialog history, while the dialog value memory keeps fixed for storing the representation of dialog context. In this way, the base model can keep track of attention history along with update-chain of decoder states, and thus generates coherent and natural responses.

KB Memory We employ a separate KB memory to encode the KB information, which is implemented with the end-to-end memory networks (Sukhbaatar et al. 2015). Each fact tuple b in the KB is represented in a triple format (*subject*, *relation*, *object*) and stored in the KB memory. The KB memory is initialized with the sum of subject and relation embeddings, and is shared across the entire conversation. We access the KB memory by K-hop reading mechanism (Sukhbaatar et al. 2015). Specifically, we use an initial query vector (decoder hidden state) as the reading head, and it loops over K hops and computes the soft attention weights at each hop. The soft memory attention decides the relevance between each memory vector and the query vector.

Response Decoder The decoder produces the target response word by word. Specifically, the t-th word in the target response is either copied from dialog value memory/KB value memory or generated from the overall vocabulary. Formally, for i-th turn, we apply a GRU to generate the target response, where the hidden state s_t at step t is learned as follows:

$$\mathbf{s}_t = \text{GRU}(\mathbf{s}_{t-1}, \mathbf{e}(y_{t-1})) \tag{2}$$

where $\mathbf{e}(y_{t-1})$ is the embedding of the previous response word y_{t-1}. We adopt reading output of dialog value memory at the last hop (see Section 3.1) as the attended dialog context $\mathbf{c}_t$. At the t-th decoding step, the word generation distribution over the vocabulary is calculated as:

$$P_v(y_t) = \text{softmax}(\mathbf{W_1}[\mathbf{s}_t; \mathbf{c}_t]) \tag{3}$$

where $\mathbf{W_1}$ is a trainable parameter.

Similarly, we use $P_d(y_t)$ and $P_{kb}(y_t)$ to represent the probabilities for copying the t-th word from the dialog memory and KB memory, respectively. A soft gate g_1 determines whether a word is copied from memories or generated from the vocabulary, and another gate g_2 determines which of the two memories is used to copy values. The final output distribution $P_\theta(y_t)$ for the t-th target word is computed as:

$$P_\theta(y_t) = g_1 P_v(y_t) + (1 - g_1)\left[g_2 P_d(y_t) + (1 - g_2)\, P_{kb}(y_t)\right] \tag{4}$$

where θ indicates the parameters of the primary network.

The dialog generation model can be trained with the training dialogues in an supervised manner. We compute the loss function $\mathcal{L}_{primary}$ as the cross-entropy between the predicted word distribution $P_\theta(y_t)$ and the ground-truth target word distribution y_t:

$$\mathcal{L}_{primary}(\theta) = -\sum_{t=1}^{T} y_t \log\left(P_\theta(y_t)\right) \tag{5}$$

where T indicates the length of the output response.

3.2 Auxiliary Reasoning Tasks

In TDSs, the target responses are guided not only by the dialog context but also by the retrieved KB entities. Most previous approaches either achieve effective KB modeling towards the KB entities extraction or a good language model for response generation, but not both. We design two novel auxiliary reasoning tasks (i.e., KB reasoning and dialog reasoning) to explicitly assist our model in reasoning over the large KB and learning dialog patterns respectively without labeling additional data manually. Each auxiliary task is also a part of the primary task.

Auxiliary KB Reasoning Retrieving accurate KB entities is critical for a task-oriented dialog system to achieve specific user goals. However, the Seq2Seq model often suffers from effectively incorporating external KB information. As revealed in previous work (Wu, Socher, and Xiong 2019), a large, dynamic external KB is equivalent to a noisy input that is difficult to encode and decode, making the generation unstable. To mitigate this problem, we propose an auxiliary KB reasoning network $P_{\phi_{kb}}$ with parameters ϕ_{kb} to facilitate the TDS retrieving accurate KB entities from external KB and integrating the extracted entities in the dialog generation. Specifically, the auxiliary KB reasoning network shares the same dialog encoder, KB memory and response decoder with the dialog generation model (primary network as illustrated in Figure 1), but without the dialog memory module. It takes the dialog history and external KB as input, and is expected to learn the corresponding meta KB knowledge from the dialog and KB. For the KB reasoning network, at each decoding step $t \in [1, T]$, the corresponding word will be generated from the vocabulary or be copied from the KB memory without considering the dialog patterns from language modeling perspective. The auxiliary KB reasoning network is optimized by minimizing the divergence between the predicted meta KB knowledge and the output of the primary network, which is defined as:

$$\mathcal{L}_{KB}(\theta, \phi_{kb}) = \sum_{t=1}^{T} \ell_{ce}\left(P_\theta(y_t), P_{\phi_{kb}}(y_t)\right) \tag{6}$$

where P_θ represents the output of the primary network with parameter θ, $P_{\phi_{kb}}$ represents the output of the KB reasoning network with parameter ϕ_{kb}, ℓ_{ce} is the standard cross-entropy function, and $\mathcal{L}_{KB}(\theta, \phi_{kb})$ is the objective function of auxiliary KB reasoning network.

Auxiliary Dialog Reasoning 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, and thus result in generating incoherent and ungrammatical responses that are dominated by words appearing with high frequency in the training data. To mitigate this issue, we propose an auxiliary dialog reasoning network P_{ϕ_d} with parameter ϕ_d, which is specialized for learning the dialog patterns so as to generate natural responses. The auxiliary dialog reasoning network shares the same dialog encoder, dialog memory and response decoder with the dialog generation model (primary network as illustrated in Figure 1), but without the KB memory module. It takes the dialog history as input, and is expected to learn the dialog patterns from the dialog. Formally, at each decoding step t, the corresponding word will be only generated from the vocabulary or copied from the dialog memory, without considering the KB knowledge. The auxiliary dialog reasoning network is optimized by minimizing the divergence between the predicted meta dialog knowledge and the output of the primary network, which is defined as:

$$\mathcal{L}_{dialog}(\theta, \phi_d) = \sum_{t=1}^{T} \ell_{ce}\left(P_\theta(y_t), P_{\phi_d}(y_t)\right) \tag{7}$$

where P_{ϕ_d} represents the output of KB reasoning network with parameter ϕ_{kb}, ℓ_{ce} is the standard cross-entropy function, $\mathcal{L}_{dialog}(\theta, \phi_d)$ is the objective function of auxiliary dialog pattern reasoning network.

3.3 Meta Cooperative Learning

As illustrated in 3, the proposed MCL consists of three networks: (i) a dialog pattern reasoning network that aims to learn the dialog patterns from the dialog context; (ii) a KB pattern reasoning network that aims to retrieve accurate entities from KB and integrate entities in dialog generation; (iii) a task-oriented dialog generation network (primary network) with the goal of not only retrieving accurate KB entities but also generating human-like responses. The primary network

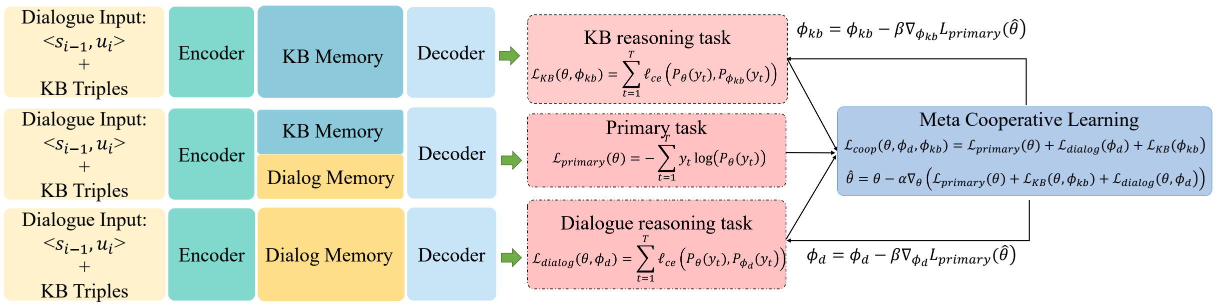

Figure 2: The overview of our model, which contains a dialog generation network (primary network), an auxiliary KB reasoning network, and an auxiliary dialog reasoning network.

is trained alongside the two auxiliary networks, with two stages per epoch. In the first stage, the primary network is trained using the ground-truth labels (training dialog) and the auxiliary labels generated by the two auxiliary networks. In the second stage, the two auxiliary networks are updated by computing their gradients with respect to the primary network's performance on the task-oriented dialog generation task. We train the three networks in an iterative manner until convergence. It is noteworthy that the auxiliary tasks are optimized by minimizing the divergence between predicted meta knowledge (output of auxiliary networks) and the output of primary network, while the primary task is optimized by ground-truth labels. The overall meta cooperative learning process is summarized in Algorithm 1.

Algorithm 1 Meta cooperative learning process.

Input: Initialized network parameters θ, ϕ_{kb} and ϕ_d; training data (X, Y).
Output: Primary network that integrates expert meta knowledge from auxiliary tasks.

```
for each epoch K do
    for each training iteration i do
        Sample a batch of training data from (X, Y)
        Update parameters of primary network:
        θ ← θ − α∇_θ (L_primary(θ) + L_KB(θ, φ_kb) + L_dialog(θ, φ_d))
    end for
    for each training iteration i do
        Sample a batch of training data from (X, Y)
        Retain one step gradient:
        θ̂ = θ − α∇_θ (L_primary(θ) + L_KB(θ, φ_kb) + L_dialog(θ, φ_d))
        Update parameters of two auxiliary networks:
        φ_kb ← φ_kb − β∇_φkb (L_primary(θ̂))
        φ_d ← φ_d − β∇_φd (L_primary(θ̂))
    end for
end for
```

Stage 1: Updating Primary Network In the first phase of each epoch, the primary network is trained using the ground-truth labels (training dialog) and the auxiliary labels generated by the two auxiliary networks. In particular, the objective function $\mathcal{L}_{primary}$ of the primary network is regularized with objective functions of the two auxiliary networks (i.e., $\mathcal{L}_{kb}$ and $\mathcal{L}_d$), inspired by the student-teacher learning (You et al. 2017). The primary network is trained to amalgamate meta KB and dialog reasoning knowledge from the two auxiliary networks respectively. Formally, the overall cooperative training objective $\mathcal{L}_{coop}$ for the primary task with cooperative learning can be defined as:

$$\mathcal{L}_{coop}(\theta, \phi_d, \phi_{kb}) = \mathcal{L}_{primary}(\theta) + \mathcal{L}_{dialog}(\phi_d) + \mathcal{L}_{KB}(\phi_{kb}) \tag{8}$$

where $\mathcal{L}_{primary}$ donates the original objective function of the primary network. $\mathcal{L}_{dialog}$ and $\mathcal{L}_{KB}$ are the objectives of two auxiliary tasks.

Stage 2: Updating Auxiliary Networks In the second stage of each epoch, the two auxiliary networks are updated by computing their gradients with respect to the primary network's performance on the task-oriented dialog generation task. Specifically, the two auxiliary tasks are updated by encouraging auxiliary labels to be chosen such that, if the primary network was to be trained using these auxiliary labels, the performance of the primary network would be maximised on this same training data. Leveraging the performance of the primary network to train the two auxiliary networks can be considered as a form of meta learning. Therefore, to update parameters ϕ_{kb} and ϕ_d of the two auxiliary networks, we define their meta objectives as follows:

$$\arg\min_{\phi_d} \mathcal{L}_{primary}\left(\widehat{\theta}\right) \tag{9}$$

$$\arg\min_{\phi_{kb}} \mathcal{L}_{primary}\left(\widehat{\theta}\right) \tag{10}$$

where $\hat{\theta}$ indicates the parameters of the primary network after one gradient update using the loss function of primary network:

$$\widehat{\theta} = \theta - \alpha \nabla_\theta \left(\mathcal{L}_{primary}(\theta) + \mathcal{L}_{KB}(\theta, \phi_{kb}) + \mathcal{L}_{dialog}(\theta, \phi_d)\right) \tag{11}$$

where α is the learning rate for the primary network.

Finally, to update parameter ϕ_{kb} and ϕ_d, we retain the computational graph of $\widehat{\theta}$ to compute gradient with respect to ϕ_{kb} and ϕ_d, respectively. Therefore, the parameters of two auxiliary networks are updated as:

$$\phi_{kb} = \phi_{kb} - \beta \nabla_{\phi_{kb}} \mathcal{L}_{primary}\left(\widehat{\theta}\right) \tag{12}$$

$$\phi_d = \phi_d - \beta \nabla_{\phi_d} \mathcal{L}_{primary}\left(\widehat{\theta}\right) \tag{13}$$

Note that since $\widehat{\theta}$ is depended on ϕ_d, ϕ_{kb} as defined in Eq. (11), the Eq. (13) and Eq. (12) are the functions of ϕ_d and

ϕ_{kb}, respectively. β is the learning rate for auxiliary tasks. Therefore, the optimization of ϕ_d and ϕ_{kb} requires gradient by gradient strategy, which is proposed in meta learning.

4 Experiments Setup

4.1 Datasets

We evaluate MCL on three benchmark TDS datasets: CamRest (Wen et al. 2016), In-Car Assistant (Eric and Manning 2017), and Multi-WOZ 2.1 (Budzianowski et al. 2018).

CamRest This corpus contains 676 multi-turn dialogs belonging to restaurant reservation domain. There are 5 turns on average per dialog (Wen et al. 2016). Following (Reddy et al. 2019), the CamRest dataset is divided into training/validation/testing sets with 406/135/135 dialogs, respectively.

In-Car Assistant The In-Car Assistant dataset (Eric and Manning 2017) is composed of 3031 multi-turn dialogs from calendar scheduling, weather information retrieval, and point-of-interest navigation domains. We divide the In-Car Assistant dataset into training/validation/testing sets with 2425/302/304 dialogs, respectively. The average number of turns per dialog is about 2.6.

Multi-WOZ This corpus extends the Multi-WOZ (Budzianowski et al. 2018) by equipping the corresponding KB to every dialog. Following the data processing in (Qin et al. 2020), there are 1,839/117/141 dialogs for training/validation/testing, belonging to restaurant, attraction, and hotel domains. There are 5.6 turns on average per dialog.

4.2 Training Details

We employ the 300-dimensional GloVe (Pennington, Socher, and Manning 2014) vectors to initialize the word embeddings. We set the size of GRU hidden units to be 256. The recurrent weight parameters are initialized as orthogonal matrices, while the other weight parameters are initialized with the normal distribution $\mathcal{N}(0, 0.01)$. We set the bias terms to be zero. We apply the Adam optimizer to train our MCL model. The learning rates α and β are initialized to $1e^{-4}$. The batch size in dialog level is set to be 8. The number of hops for the dialog memory and KB memory is set to be 3. The embedding dimension of memory network is set to be 256. We set the dropout rate to be 0.2.

4.3 Baselines

We compare the proposed MCL model with several strong end-to-end TDSs: (1) **Seq2Seq/+Attn** that adopts seq2seq with and without attention (Luong, Pham, and Manning 2015); (2) **Ptr-Unk** that adopts Seq2Seq with a copy mechanism to copy unknown words during generation (Gulcehre et al. 2016); (3) **Mem2Seq** that adopts a memory network with multi-hop attention for attending over dialog history and KB triples (Madotto, Wu, and Fung 2018); (4) **BossNet** that adopts a bag-of-sequences memory network to disentangle language model from KB in TDS (Raghu, Gupta et al. 2019); (5) **DSR** that uses dialog state representation to retrieve the KB implicitly (Wen et al. 2018); (6) **ECET** that adopts a two-step strategy to retrieve KB entities by firstly retrieving the most relevant KB and then locating the most relevant KB column via attention (Qin et al. 2019); (7) **MLM** that uses a multi-level memory network to model KB tuples and dialog context separately (Reddy et al. 2019); (8) **GLMP** that employs a memory network with a global memory pointer and a local memory pointer to strengthen the copy ability (Wu, Socher, and Xiong 2019).

Model	BLEU	Entity F1
Seq2Seq	7.9	17.6
Seq2Seq+Attn	7.7	21.4
Ptr-Unk	5.1	16.4
Mem2Seq	13.5	33.6
BossNet	15.2	43.1
MLM	16.1	55.2
ECET	<u>18.5</u>	<u>58.6</u>
GLMP	16.7	50.6
MCL (Ours)	**20.1**	**59.2**

Table 2: Automatic evaluation results on CamRest.

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
MLM	<u>15.6</u>	55.5	67.4	<u>54.8</u>	45.1
ECET	14.1	53.7	54.5	52.2	<u>55.6</u>
GLMP	14.8	<u>60.0</u>	<u>69.6</u>	**62.6**	53.0
MCL	**17.2**	**60.9**	**70.6**	**62.6**	**59.0**

Table 3: Automatic evaluation results on In-Car Assistant.

4.4 Automatic Evaluation Metrics

Similar to (Wu, Socher, and Xiong 2019), we evaluate MCL and baseline methods with two automatic evaluation metrics: BLEU (Papineni et al. 2002) and entity F1 (Madotto, Wu, and Fung 2018).

- **BLEU**: BLEU measures the n-gram (i.e., 4-gram) overlap between the produced responses and the gold responses. BLEU is a popular metric to measure the TDS's ability to accurately generate the dialog from the language model perspective.
- **Entity F1**: We utilize the entity F1 score to measure the system's capability of generating relevant entities to accomplish certain tasks by retrieving accurate entities from the provided KB. There are a set of reference entities for each utterance. The entity F1 score is computed by micro-averaging the precision and recall over KB entities of the generated responses.

5 Experimental Results

5.1 Automatic Evaluation Results

We use the response generated by MCL as the final output response. The automatic evaluation results on the three datasets (i.e., CamRest, In-Car Assistant, Multi-WOZ) are reported in Tables 2-4, respectively. From Tables 2-4, we can observe that MCL substantially outperforms baseline models by a noticeable margin on both BLEU and Entity F1. MCL outperforms the strong baseline GLMP by about 5% on BLEU and 9% on entity F1, verifying the effectiveness of our model

Model	BLEU	Ent.F1	Res.F1	Att.F1	Hot.F1
Seq2Seq	4.3	9.2	10.5	8.7	8.2
Seq2Seq+Attn	4.5	11.6	11.9	10.8	11.1
Ptr-Unk	4.8	17.4	19.6	16.6	15.5
Mem2Seq	6.6	21.6	22.4	22.0	21.0
BossNet	5.7	25.3	26.2	24.8	23.4
MLM	9.2	27.8	29.8	27.4	25.2
DSR	9.1	30.0	33.4	28.0	27.1
GLMP	6.9	32.4	**38.4**	24.4	28.1
MCL	**13.6**	**32.6**	34.4	**30.2**	**29.8**

Table 4: Automatic evaluation results on Multi-WOZ 2.1.

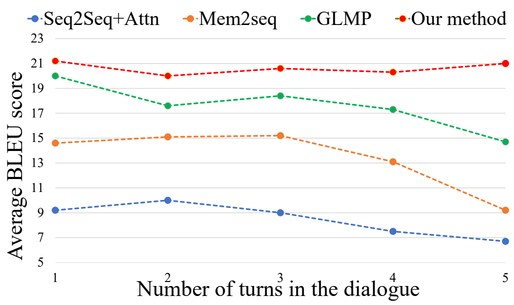

Figure 3: The BLEU scores of MCL and several baselines with the increase of dialog turns on the CamRest dataset.

in single domain task-oriented dialog generation. Similar trends of improvement are observed on In-Car Assistant and Multi-WOZ 2.1 with multi-domain dialogs. MCL achieves significantly better BLEU and entity F1 scores than the compared methods. In particular, on the Multi-WOZ 2.1 dataset, MCL outperforms the best baselines (MLM and GLMP) by 48% on BLEU score. The advancement is mainly benefited from the two auxiliary reasoning tasks that learn implicit meta knowledge derived from the KB and dialog patterns.

We also investigate the stability of different TDSs with the increase of dialog turns. Figure 3 shows the changes in average BLEU scores of MCL and baselines along with the increase of dialog turns on CamRest. In particular, the BLEU scores of Seq2Seq-Attn and Mem2Seq begin to decrease after three turns. The results of all baseline models deteriorate sharply after four dialog turns. While MCL achieves stable performance even in the last few turns, which demonstrates the effectiveness of MCL in reasoning over the dialog history, KB knowledge, and historical inference process.

5.2 Human Evaluation Results

Following previous study (Wu, Socher, and Xiong 2019), we also evaluate MCL with human annotation by taking both informativeness (*Infor.*) and human-likeness (*Humanlike.*) of the generated responses into consideration. We randomly select 100 dialogs from test sets of the experimental data, and invite three NLP researchers to assign each response a score (1-5) for *Infor.* and *Humanlike.*, respectively. The agreement ratios computed with Fleiss' kappa (Fleiss 1971) are 0.57 on CamRest, 0.49 on In-Car Assistant, and 0.60 on Multi-WOZ, showing moderate agreement. Table 5 reports the average rating scores over all the annotators. From Table 5, we can observe that MCL outperforms baselines in terms of both informativeness and human-likeness by a noticeable margin,

Model	CamRest		In-Car		MultiWOZ 2.1	
	Infor.	HL	Infor.	HL	Infor.	HL
Mem2Seq	3.41	3.77	3.75	3.67	3.12	3.29
BossNet	3.64	4.01	3.76	3.79	3.34	3.35
GLMP	4.07	4.14	4.21	4.10	4.01	3.91
MCL	**4.19**	**4.28**	**4.22**	**4.18**	**4.12**	**4.01**

Table 5: Human evaluation results on CamRest, In-Car Assistant and Multi-WOZ 2.1. HL stands for Humanlike.

Model	CamRest		In-Car		Multi-WOZ	
	BLEU	Ent. F1	BLEU	Ent.F1	BLEU	Ent.F1
MCL (ours)	20.1	58.2	17.2	60.6	13.6	32.6
w/o ML	20.0	55.2	15.2	56.7	10.3	30.5
w/o AK	17.6	55.0	16.7	54.4	10.1	27.3
w/o AD	15.7	57.3	15.7	60.3	9.7	31.7
w/o AK+AD	16.3	53.0	14.2	53.8	9.1	29.7

Table 6: Ablation results of our MCL model on three datasets.

which is consistent with the automatic evaluation.

5.3 Ablation Study

For the purpose of analyzing the effectiveness of different components of MCL, we conduct an ablation test of MCL by removing the meta learning (denoted as w/o ML), the auxiliary KB reasoning task (w/o AK), the auxiliary dialog reasoning task (w/o AD), and both auxiliary reasoning tasks (w/o AK+AD). In particular, when removing the meta learning, we simply pre-train the KB and dialog reasoning networks with training data without updating them in the training process of the primary task. To create the training data for the two auxiliary tasks, we use a special token "NEN" to mask all non-entity words in the gold response for the auxiliary KB reasoning task, and use a special token "ENT" to mask all KB entities words in the reference responses for auxiliary dialog reasoning task.

The ablation test results are demonstrated in Table 6. Generally, both auxiliary KB and dialog reasoning tasks contribute great performance improvement to MCL. This is within our expectation since the KB reasoning task helps our model extract more accurate KB entities and the dialog reasoning task assists MCL in learning dialog patterns from dialog context. It is no surprise that removing meta learning leads to large performance degradation on all three datasets. This verifies that the meta learning can automatically learn the optimal auxiliary tasks.

6 Conclusion

In this paper, we proposed a meta cooperative learning framework to improve the performance of TDSs in retrieving accurate KB entities and generating natural responses simultaneously by amalgamating meta knowledge from the auxiliary KB and dialog reasoning tasks without requiring access to any further data. We conducted extensive experiments on three benchmark datasets. The experimental results demonstrated that our model achieves impressive results compared to the state-of-the-art TDSs. Interestingly, TDS with MCL outperformed the single-task learning, even though the three networks adopted the same amount of training data.

Acknowledgments

Min Yang was partially supported by the National Natural Science Foundation of China (NSFC) (No. 61906185), Shenzhen Basic Research Foundation (No. JCYJ20200109113441941), Youth Innovation Promotion Association of CAS China (No. 2020357). Chengming Li was partially supported by Shenzhen Basic Research Foundation (No. JCYJ20180302145607677).

References

Bordes, A.; Boureau, Y.-L.; and Weston, J. 2017. Learning End-to-end Goal-oriented Dialog. *ICML* .

Budzianowski, P.; Wen, T.-H.; Tseng, B.-H.; Casanueva, I.; Ultes, S.; Ramadan, O.; and Gašić, M. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *EMNLP* .

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

Chen, X.; Xu, J.; and Xu, B. 2019. A Working Memory Model for Task-oriented Dialog Response Generation. In *ACL*, 2687–2693.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* .

Eric, M.; Krishnan, L.; Charette, F.; and Manning, C. D. 2017. Key-Value Retrieval Networks for Task-Oriented Dialogue. In *SIGdial*, 37–49.

Eric, M.; and Manning, C. D. 2017. A Copy-augmented Sequence-to-sequence Architecture Gives Good Performance on Task-oriented Dialogue. *EACL* .

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400* .

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

Gulcehre, C.; Ahn, S.; Nallapati, R.; Zhou, B.; and Bengio, Y. 2016. Pointing the unknown words. In *ACL*, 140–149.

He, W.; Sun, Y.; Yang, M.; Ji, F.; Li, C.; and Xu, R. 2021. Multi-goal multi-agent learning for task-oriented dialogue with bidirectional teacher–student learning. *Knowledge-Based Systems* .

He, W.; Yang, M.; Yan, R.; Li, C.; Shen, Y.; and Xu, R. 2020. Amalgamating Knowledge from Two Teachers for Task-oriented Dialogue System with Adversarial Training. In *EMNLP*, 3498–3507.

Hospedales, T.; Antoniou, A.; Micaelli, P.; and Storkey, A. 2020. Meta-learning in neural networks: A survey. *arXiv preprint arXiv:2004.05439* .

Lei, W.; Jin, X.; Kan, M.-Y.; Ren, Z.; He, X.; and Yin, D. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *ACL*.

Luong, M.-T.; Pham, H.; and Manning, C. D. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*, 1412–1421.

Madotto, A.; Lin, Z.; Wu, C.-S.; and Fung, P. 2019. Personalizing Dialogue Agents via Meta-Learning. In *ACL*, 5454–5459.

Madotto, A.; Wu, C.-S.; and Fung, P. 2018. Mem2Seq: Effectively Incorporating Knowledge Bases into End-to-End Task-Oriented Dialog Systems. In *ACL*, 1468–1478.

Mi, F.; Huang, M.; Zhang, J.; and Faltings, B. 2019. Meta-learning for low-resource natural language generation in task-oriented dialogue systems. In *IJCAI*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*, 311–318.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. GloVe: Global Vectors for Word Representation. In *EMNLP*.

Qin, L.; Liu, Y.; Che, W.; Wen, H.; Li, Y.; and Liu, T. 2019. Entity-Consistent End-to-end Task-Oriented Dialogue System with KB Retriever. In *EMNLP/IJCNLP*.

Qin, L.; Xu, X.; Che, W.; Zhang, Y.; and Liu, T. 2020. Dynamic Fusion Network for Multi-Domain End-to-end Task-Oriented Dialog. In *ACL*.

Raghu, D.; Gupta, N.; et al. 2019. Disentangling Language and Knowledge in Task-Oriented Dialogs. In *NAACL-HLT*, 1239–1255.

Reddy, R.; Contractor, D.; Raghu, D.; and Joshi, S. 2019. Multi-Level Memory for Task Oriented Dialogs. In *NAACL-HLT*, 3744–3754.

Sukhbaatar, S.; Weston, J.; Fergus, R.; et al. 2015. End-to-end memory networks. In *NeurIPS*, 2440–2448.

Thrun, S.; and Pratt, L. 1998. Learning to learn: Introduction and overview. In *Learning to learn*, 3–17. Springer.

Wang, J.; Liu, J.; Bi, W.; Liu, X.; He, K.; Xu, R.; and Yang, M. 2020. Dual Dynamic Memory Network for End-to-End Multi-turn Task-oriented Dialog Systems. In *COLING*.

Wen, H.; Liu, Y.; Che, W.; Qin, L.; and Liu, T. 2018. Sequence-to-Sequence Learning for Task-oriented Dialogue with Dialogue State Representation. In *COLING*, 3781–3792.

Wen, T.-H.; Gasic, M.; Mrkšić, N.; Barahona, L. M. R.; Su, P.-H.; Ultes, S.; Vandyke, D.; and Young, S. 2016. Conditional Generation and Snapshot Learning in Neural Dialogue Systems. In *EMNLP*, 2153–2162.

Wu, C.-S.; Socher, R.; and Xiong, C. 2019. Global-to-local Memory Pointer Networks for Task-Oriented Dialogue. In *ICLR*.

Yan, M.; Zhang, H.; Jin, D.; and Zhou, J. T. 2020. Multi-source Meta Transfer for Low Resource Multiple-Choice Question Answering. In *ACL*, 7331–7341.

You, S.; Xu, C.; Xu, C.; and Tao, D. 2017. Learning from Multiple Teacher Networks. In *SIGKDD*, 1285–1294.