Uni-ETOD: User-Need-Driven Chain of Thought Framework for Fully End-to-end Task-oriented Dialogue System

Anonymous ACL submission

Abstract

Fully End-to-End Task-Oriented Dialogue Systems (Fully ETOD) retrieve knowledge from a knowledge base in a differentiable manner and generate responses using a language model generator without the need for modular training. However, Fully ETOD faces some challenges. During the retrieval process, the retriever retrieves the knowledge base in a black-box manner, making it difficult for the generator to differentiate the large amount of knowledge obtained by the retriever. This leads to a degradation in the quality of the responses and the trustworthiness of the system. Moreover, as the size of the knowledge base grows, it may exacerbate the risk of this problem. To address this challenge, we first design a dataset for Fully ETOD based on large-scale knowledge bases called FakeRest to solve the scarcity of annotated dialogue data based on large-scale knowledge bases. We also propose a User-need-driven Chain of Thought Framework (Uni-ETOD) for Fully ETOD, which aims to guide LLMs to gradually understand users' thought processes and improve the quality of responses in Fully ETOD. We use ChatGPT, Gemini, Llama3, Mistral, and ChatGLM as the backbone models of the system. On FakeRest, we comprehensively evaluate the capability of each step of Uni-ETOD. The results show that Uni-ETOD will help LLMs better distinguish the retrieved knowledge and enhance the credibility and interpretability of the whole system.

1 Introduction

Task-oriented dialogue (TOD) systems can accomplish specific tasks, such as booking restaurant reservations and providing transportation navigation, through user interaction and leveraging an external knowledge base. Traditional TOD systems follow a pipeline approach and consist of four interconnected modular components (Qin et al., 2020; Jacqmin et al., 2022; Hosseini-Asl et al., 2020).

Figure 1: A sample demonstration of Fully ETOD. In the first round, the system did not select all the entities that met the needs. In the second round, the system selected irrelevant attributes for the response. We mark the correct entities and attributes in green. Ignored or incorrect entities and attributes are labeled in red.

The Modularly End-to-End Task-oriented Dialogue System (Modularly EToD) trains all components in an end-to-end manner while optimizing their parameters. In contrast to traditional TOD and Modularly EToD, the Fully Task-oriented End-to-End Dialogue System (Fully ETOD) (Eric and Manning, 2017) encodes knowledge bases (KBs) and uses neural networks to query the KBs in a differentiable manner. Fully ETOD generates a system response directly given only the dialogue history and the corresponding knowledge base. Therefore, it has received great attention from both academia and industry. Although Fully ETOD exhibits excellent data scalability, efficiently retrieving the desired entities and attributes poses a challenge due to the abundance of irrelevant knowledge in the retrieved results. This issue hampers the generator's ability to distinguish between retrieved entities effectively and extract valid information.

Existing Fully ETODs tend to follow the retrieval-generation paradigm. Retrieval-augmented generation (RAG) (Lewis et al., 2020; Ren et al., 2021; Singh et al., 2021) improves the quality and relevance of the generated text and achieves positive results in knowledge-intensive tasks. Q-TOD (Tian et al., 2022) efficiently implements retrieval-enhanced generation on Fully ETOD, thus alleviating the domain adaptation problem. MK-TOD (Shen et al., 2023) raises the problem of mismatch between the retrieval and generation processes, and incorporates a variety of meta-knowledge to guide the generator to increase the retrieval utilization of the results. However, for the generator, the retrieval process of the retriever remains a black-box process. This means that although a retriever can provide retrieval results with high recall, the results still contain a large number of irrelevant entities and attributes. It is difficult for the generator to differentiate between these retrieved entities and to select the attributes of the entities that meet the needs, as shown in Figure 1. This is referred to as the low precision problem of the retrieval process. Meanwhile, the black-box and low-precision retrieval process is difficult to analyze, which damages the user's trust (Qin et al., 2023). We consider the low precision problem and non-interpretability of the retrieval process as bottlenecks in the existing Fully ETOD. Another core challenge with existing Fully ETOD is the lack of annotated dialogue data based on large-scale knowledge bases (Qin et al., 2023). Existing Fully ETOD datasets are often based on small knowledge bases or modified from existing datasets. Due to not being tailored for large-scale datasets, existing Fully ETOD datasets frequently encounter discrepancies between responses and annotated knowledge.

Recent developments in large language modeling provide us with solutions to the above problems. Our paper introduces a **U**ser-**n**eed-dr**i**ven Chain of Thought Framework for Fully ETOD (Uni-ETOD). Uni-ETOD aims to improve the precision and interpretability of the retrieval process, and thus the quality of the responses. The framework consists of three steps: 1) Retrieval Based on User Needs: Based on the dialogue context, LLMs will generate the user's needs and use the embedding model to retrieve the most relevant knowledge in a large-scale knowledge base. 2) Knowledge Refinement: Based on the retrieval results obtained by the retriever, the LLMs will filter the entities and attributes that match the user's needs, resulting in a more accurate retrieval result. The results can be directly displayed to the user as part of the response. 3) Response Based on Refined Knowledge: Based on the retrieval results with higher precision, the generator will make a more credible response.

In addition, we propose an automated method for constructing dialog data based on a large-scale knowledge base. We utilize LLMs to simulate restaurant scenarios and construct FakeRest. FakeRest is specifically tailored for Fully ETOD of large-scale knowledge bases, containing detailed annotated data to enhance the knowledge base retrieval capability of LLMs. Refer to chapter 3 for more details.

We apply Uni-ETOD to several LLMs, including two closed-source models, ChatGPT (Brown et al., 2020), Gemini (Team et al., 2023), and three open-source models, Llama3 (AI@Meta, 2024), Mistral (Jiang et al., 2023), and ChatGLM (Du et al., 2022). On the FakeRest dataset, we utilize the task to evaluate the enhancement of Uni-ETOD on the retrieval process and the response process, respectively. The experimental results show that Uni-ETOD can effectively alleviate the low precision problem in the retrieval process, and enhance the quality of responses and interpretability.

2 Related Work

2.1 Fully End-to-End Task-oriented Dialog

We use whether or not we query KBs as APIs using beliefs as a criterion to differentiate between modular ETODs and Fully ETODs. Fully ETODs retrieve knowledge bases in a differentiable way. We classify existing Fully ETODs as being categorized into two types. First, the knowledge base is stored in the model parameters, and the system retrieves it implicitly and generates a response to the user. GPT-KE (Madotto et al., 2020) learns knowledge base embedding through data augmentation and responding to users. ECO (Huang et al., 2022) guides the response generation in the end-to-end system by autoregressively generating entities. However, such an approach mixes the retrieval and response processes and inevitably generates low-confidence generation results, especially when the size of the knowledge base becomes large.

Second, the system explicitly retrieves KBs through a retriever. DialoKG (Rony et al., 2022) selects relevant triples by graph embedding. Q-TOD (Tian et al., 2022) utilizes a rewritten query in combination with the RAG technique to improve retrieval performance. MAKER (Wan et al., 2023) queries entities and attributes separately through two retrievers. MK-TOD (Shen et al., 2023) mitigates the misalignment between the retriever and the generator by combining meta-knowledge. Although retrieving the KB explicitly provides the retrieved results compared to the first approach, the black-box retrieval process still limits the interpretability. Moreover, such a retrieval process will inevitably introduce a large number of irrelevant entities and attributes. Our work aims to utilize the power of LLMs to alleviate the interpretability and low precision problems of the retrieval process.

2.2 Large Language Models for ETOD

Recently, LLMs have achieved great success and demonstrated extraordinary text generation and reasoning capabilities (Suzgun et al., 2023; Pu and Demberg, 2023; Kojima et al., 2022). Unlike small language models that have difficulty solving complex problems, LLMs can solve complex problems with various prompting strategies (Zhao et al., 2023). This is based on the amazing ability that LLMs show in multi-hop reasoning. Wei et al. (Wei et al., 2022)investigated the Chain of Thought (CoT) prompting technique in LLMs by inducing the model to generate intermediate steps to improve the precision of answers. Meanwhile, based on the powerful contextual learning capability of LLMs, many existing works combine LLMs with traditional TOD systems. They generally use the zero-shot or few-shot approach to explore the capability of LLMs applied to individual modules (Pan et al., 2023; Heck et al., 2023; Hudeček and Dušek, 2023; Parikh et al., 2023). However, there is a gap in the work on applying LLMs to Fully EToD. The lack of a Fully ETOD dataset and training paradigm based on a large-scale knowledge base is a hindrance.

3 FakeRest: A Fully ETOD Dataset for the Large-Scale Database

3.1 Why build the FakeRest dataset?

To address the challenge of scarce labeled data faced by Fully ETOD, we propose a construction method and construct FakeRest, a dataset of scheduled restaurant scenarios designed for Fully ETOD based on a large-scale knowledge base. We utilize LLMs to simulate the dialog scenarios between the user and the system in the restaurant. Based on a predefined user need path, the user LLM and the system LLM will have multiple rounds of dialog until the system finds (or fails to find) the only restaurant that matches the user's need. In this process, we will record the thought and retrieval process of the user and the system in detail as annotated data.

FakeRest has the following advantages:

1) **Based on Large-Scale Knowledge Base:** Unlike the existing Fully ETOD datasets, which are based on a small-scale knowledge base. FakeRest's knowledge base contains entities for 120 different restaurants. Similar to the format of CamRest (Rojas-Barahona et al., 2016), each restaurant in the knowledge base contains seven attribute values. Four are private attributes (name, phone, address, postcode) for each restaurant, and the values of the private attributes are completely different. Including three public attributes (area, food, price range), the public attributes can match the user's needs.

2) **Data Consistency:** The existing Fully ETOD datasets annotate the corresponding knowledge base based on the dialogue content of the original dataset. This could lead to inconsistencies between the dialogue content and the annotated knowledge base, or the system may fail to respond based on all the entities that match the user's needs. In contrast, during each dialogue round in FakeRest, the service system will retrieve all the restaurants that match the user's needs from the knowledge base of 120 restaurants and respond to the user. Such an approach ensures the consistency of the annotated knowledge and responses.

3) **More Detailed Annotation:** In addition annotating the user's and system's utterances, we provided detailed annotations of the user's and system's thought processes, including the user's needs, all entity IDs, and attribute values that aligned with the user's needs for the round. To maintain data diversity and balance, we established various need paths and target restaurants for each dialogue. We also created scenarios where no restaurants in the knowledge base matched the user's needs.

3.2 How to build the FakeRest dataset?

We propose a method to automatically construct dialog data for large-scale knowledge bases and make an attempt with a subscription restaurant scenario. First, we construct a knowledge base of 120 restaurants using ChatGPT (Brown et al., 2020). Then, we designed different need paths based on different attributes of the restaurants in the knowledge base. Each need path corresponds to one LLM user. We design 900 users, among which 720 users found the needed restaurants and 180 users did not. Finally, based on the need paths, we use templates to construct query statements to find all the entities and attributes that match the user's needs, and as part of the annotations. The LLM user and the LLM system will generate multiple rounds of dialogs based on these annotations. We describe these three steps in detail next. Please refer to the Appendix A for prompts.

Constructing the Knowledge Base We defined 18 public attributes (5 area, 10 food, 3 price range) and simulated 120 restaurants based on different combinations of public attributes. It is worth noting that we could have simulated 150 restaurants with slightly different public attributes. However, in order to generate scenarios that did not align with the user's needs, we randomly removed 30 restaurants. Subsequently, we used ChatGPT to create unique private attributes for each restaurant, which were then reviewed and refined manually. As a result, we obtained 120 distinct restaurants.

Constructing the Need Path Similar to the process of constructing restaurants, we construct the user's need path by combining various public attributes. We use food as the starting point of the path and choose either area or price range as the subsequent step. For instance, if the user is looking for a Korean restaurant in the eastern part of the city, the need path for this round is [Korean(food), east(area)]. Each path will end in the final round by finding the only restaurant that meets the need or by not finding any suitable restaurant. Once the restaurant is identified, the need path will take the private attribute as the next step (e.g., postcode, phone). We assign the corresponding LLM users based on the 900 need paths and utilize the interactions between each user and the system as training data.

Constructing Detailed Annotated Data We transform the need path into a fluent sentence as a user need and allow the LLM user to express the current need based on the dialog context and the user need. For instance, "You are looking for a Korean restaurant in the east area.". Unlike the subsequent methods that utilize RAGs for retrieval, we convert the need path into a deterministic search statement to search a list of all restaurants in the knowledge base that meet the need. For more precise labeling, we label the IDs of the retrieved entities and the values of the attributes of the current round of needs, rather than all the entity information. For example, {"name": "taste discoveries", "food": "korean", "area": "east"}. The system will respond to the LLM user with detailed labeled information based on the context of the dialogue.

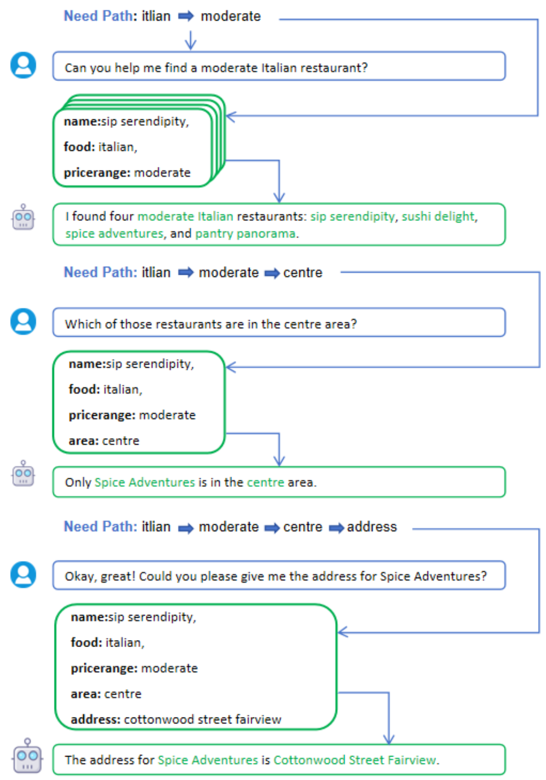

Figure 2: Figure of the process of constructing FakeRest. Based on the set user's need path, the user LLM and system LLM simulate the restaurant reservation scenario.

3.3 Dataset Statistics

In Table 1 we summarize the statistics for FakeRest. We use the partitioning of training/test, where #Entities denotes the number of entities in the knowledge base and #Attributes denotes the number of attribute values in the knowledge base.

Dataset	#Dialogues	#Utterances	#Entites	#Attributes
FakeRest	765/135	4252/752	120	498

Table 1: Dataset statistics of the FakeRest dataset.

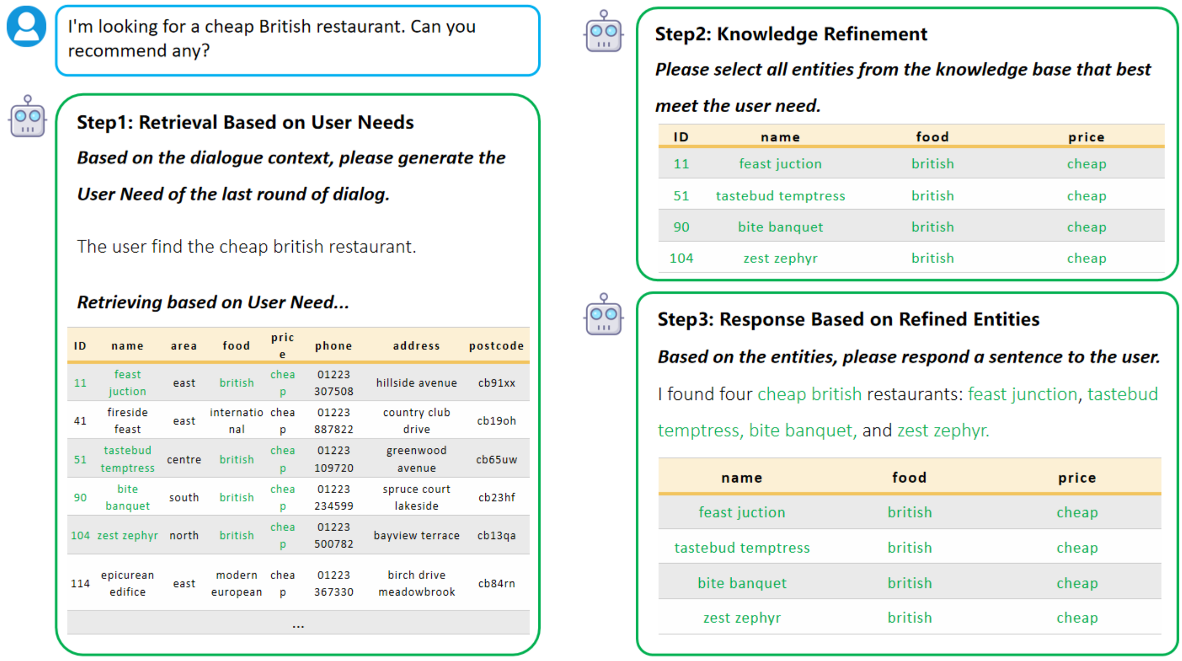

Figure 3: An illustration of a Uni-ETOD. In each round of dialogue, Uni-ETOD retrieves and refines knowledge according to the user's needs. To better represent the knowledge base information, we use a graph instead of a JSON list. Finally, Uni-ETOD will be able to return structured knowledge that is interpretable after refinement (e.g., a list of restaurants that meet the user's needs) and a paragraph of highly reliable response.

4 Uni-ETOD: User-Need-Driven Chain of Thought Framework for Fully ETOD

The Fully ETOD task can be defined as given a dialog history H and an associated knowledge base KB. H = $(u_1, s_1), (u_2, s_2),..., (u_{n-1}, s_{n-1}), u_n$, where u_n and s_n denote the n-th round of user utterances and system utterances, respectively. The KB = $(e_1,e_2, ... ,e_m)$, e_m=$(a_1,a_2,... ,a_k)$, where e_m and a_k denote entities and attributes in the knowledge base. The purpose of the system agent is to predict the system response s_n, denoted as S.

In recent years, LLMs have changed the paradigm of natural language, showing strong multi-hop reasoning capabilities (Zhao et al., 2023). Inspired by CoT (Wei et al., 2022), we consider replacing the traditional retrieval-generation paradigm with a multi-step reasoning framework. We aim to assist LLMs to understand the user's needs step by step and explain an understandable retrieval process based on the user's needs. The retrieval results will show which entities and attribute values match the user's needs and ultimately provide a high-quality response. We will detail the three steps of Uni-ETOD and the required prompt templates.

4.1 Retrieval Based on User Needs

In this step, we first construct a dataset corresponding to user needs and entities in the knowledge base using the training set of FakeRest. Since FakeRest has detailed annotations, we construct a sentence pair dataset of user needs and entities for fine-tuning the embedding model. The dataset can be represented as $D = (un_1, [e_1, e_2, ..., e_i]), ..., (un_i, [e_1, e_2, ..., e_j])$, where un and e represent the user's needs and the entities in the knowledge base, respectively. We train our embedding model using all positive samples to compute similarity scores and cross-entropy loss of labels. See the Appendix B for experimental details.

Then we use LLMs to summarize the user need based on the dialogue history H. We use the following template:

> Based on the 'Dialog Context', please generate the 'User Need' of the 'Last Round' of dialog.
> Dialog Context: [Dialogue History H]
> Last Round:[u_n]

Given the dialogue history H, we allow the LLMs to summarize the user's needs for the last round of dialogue u_n. This step can be represented as Equation 1, where U is a concise sentence representing the user's needs.

$$U = argmax\ P\left(y|H, u_n\right) \quad (1)$$

Finally, we use the U to retrieve the top-k entities in the knowledge base that match the user's needs, which can be represented as:

$$E_{top-k} = P\left(U, KB\right) \quad (2)$$

4.2 Knowledge Refinement

In this step, we allow the LLMs to refine the retrieved knowledge based on the user's needs. The LLMs filter out entities and attributes that align with the user's needs from the retrieved knowledge. This step is designed to address the low precision problem of the retrieved results, aiming to enhance the credibility of the retrieved knowledge. Furthermore, we allow LLMs to provide additional explanations for the process and present high-precision knowledge to the user, improving the system's interpretability. We follow the template below:

> Please select all 'Entities' from the 'Knowledge Base' that best meet the user needs.
> Dialogue Context: [Dialogue History H]
> Knowledge Base: [Retrieved Entities E]

Given the dialog history H and the retrieval result E from the previous step, we guide the LLMs to select the knowledge that meets the user's needs. The refined knowledge contains only the entities and attributes that match the user's needs. This step can be represented as Equation 3, where $E_{refined}$ is the set of refined entities.

$$E_{refined} = argmax\ P\left(y|H, E\right) \quad (3)$$

4.3 Response Based on Refined Knowledge

In this step, LLMs provide high quality responses to customers based on retrieval results with high precision and recall. We use the following template:

> Based on the 'Entities', please respond a sentence to the user.
> Dialog Context: [Dialog History H]
> Entities: [Refined Entities $E_{refined}$]

Given the dialog history H and the refined retrieval entities $E_{refined}$, we guide the LLMs to generate the final response. This step can be represented as Equation 4. S represents the final system response.

$$S = argmax\ P\left(y|H, E_{refined}\right) \quad (4)$$

5 Experiments

We first explain the problem of low precision in the retrieval process. To comprehensively evaluate the effectiveness of Uni-ETOD, we will assess the retrieval and response processes of Fully ETOD separately.

5.1 Low Precision of the Retrieval Process

As the scale of the knowledge base increases, it becomes increasingly challenging for the retriever to find all the entities that match the user's needs from a large number of entities. Many works enhance the recall of the retrievers through more parameterized encoders and sophisticated training methods. However, the retrieval results inevitably contain numerous irrelevant entities and attributes. Since the entities in the knowledge base are often very similar and there are usually very few entities that match the user's needs, this results in low precision problems in the retrieval process. These irrelevant entities and attributes will be directly utilized as the basis for the generator to make a response, which will destroy the credibility of the response.

Shen et al. (Shen et al., 2023) proposed the retrieval-generation misalignment problem, which refers to the inconsistency between the recall rate of the retrieval process and the response quality. We believe that the problem of low precision in retrieval results is the main reason for this phenomenon. This is because retrieval results that contain a large number of irrelevant entities will be more dependent on the generator's capabilities. Additionally, as illustrated in Table 2, even a high recall retriever exacerbates the low precision problem when increasing the k value to handle larger retrievals, impacting the response precision. The low precision problem may hinder the development of Fully ETOD based on large-scale knowledge bases, as more efficient retrievers may not effectively address this issue. Our approach addresses this problem by leveraging the reasoning abilities of LLMs through stepwise reasoning. In our results, Uni-ETOD notably enhances the precision of retrieval results and demonstrates improved alignment between the retrieval and generation processes.

k	Precision@10	Recall@10	Entity Precision	Entity Recall
5	27.5	80	63.9	70.3
10	16.1	94	63.7	70.5
15	11.1	97.2	62.1	73.4
20	8.5	99.1	60.1	73.3

Table 2: Trends in precision and recall during retrieval and response as k increases

5.2 Overall Results on Retrieval Process

We used bge-large-en-v1.5 (Xiao et al., 2023) as the base model and also compared the retrieval results of an embedded model that utilizes fine-tuning of user needs (without Knowledge Refinement) as well as UniETOD. To evaluate the retrieval capability of Fully ETOD, the recall of the retrieval process is evaluated in addition to Recall@k (Shen et al., 2023). We also propose Precision@k and F1@k to evaluate the precision and comprehensive performance of the retrieval process.

Model	Precision@10	Recall@10	F1@10
BGE-Large	10.4	60.4	17.7
BGE-Large+User Need	16.9	98.9	29.0
Uni-ETOD(ChatGLM)	79.4	85.9	82.5
Uni-ETOD(Llama3)	96.0	96.7	96.4
Uni-ETOD(Mistral)	97.2	94.1	95.7
Uni-ETOD(Gemini)	63.2	92.9	75.3
Uni-ETOD(ChatGPT)	70.0	89.0	78.3

Table 3: Experimental results of the retrieval process on FakeRest dataset.

As shown in Table 3, we observe that the fine-tuned embedding model, based on FakeRest's detailed annotation, performs well in achieving very high recall results for short user needs. The fine-tuned embedding model enhances recall by 38.5 percent under the top-10 retrieval setting. However, the retrieval process is hindered by the low precision problem due to the constraints of the large-scale knowledge base. This problem is well mitigated by Uni-ETOD, which improves precision at the expense of a minimal reduction in recall. For instance, Uni-ETOD (Llama3) decreases the recall by 2.2 percent but improves the precision by 79.1 percent. Although ChatGPT and Gemini were unable to fine-tune the Knowledge Refined step, both models still improved the precision of the retrieval results in the zero-shot setting. The results demonstrate that Uni-ETOD improves the precision of the retrieval process, thereby elevating the overall quality of Fully ETOD. Moreover, the user-need-based retrieval process in Uni-ETOD notably enhances the recall of the base model, achieving 98.9 percent at the top-10 setting.

5.3 Overall Results on Response Process

In this section, we show all the implementation details and all the experimental results of the LLMs in the response process.

5.3.1 Implementation Details

We use 5 LLMs for our experiments, including 2 closed-source models and 3 open-source models. Specifically, we use GPT3.5 (gpt-3.5-turbo) (Brown et al., 2020) from the OpenAI API and Gemini (gemini-1.5-flash) (Team et al., 2023)] from Google Gemini API. Additionally, we include three open-source models in our experiments: ChatGLM3 (chatglm3-6B) (Du et al., 2022) and Llama3 (meta-llama3-8B-instruct) (AI@Meta, 2024) and Mistral (mistral-7B-instruct-v0.2) (Jiang et al., 2023). The temperature is set to 0.1, while all other hyperparameters are set to default values.

We use LoRA (Hu et al., 2021) to fine-tune all LLMs. The LLMs are fine-tuned on a 24G NVIDIA 3090. We set training batch size to 1, epoch number to 3, learning rate to 5e-5, and warm-up steps to 20.

We compare the improvement of Uni-ETOD on the overall performance and reliability of LLMs in the response process by using retrieve-generate paradigm (base) and retrieve+zero-shot CoT (zeroCoT) (Kojima et al., 2022) as a baseline. To evaluate the quality of Fully ETOD's responses, we use the BLEU (Papineni et al., 2002), Entity F1 (Eric and Manning, 2017) metrics to assess the consistency of responses and the generator's ability to respond with correct knowledge.

5.3.2 Results on Zero-shot Reasoning and Supervised Fine-tuning

Since Gemini and ChatGPT are not fine-tunable, we show performance with zero shots as well as results with the fine-tuned Llama3 as the knowledge refinement component(+KR). On Llama3, ChatGLM, and Mistral we show performance with zero-shot inference and fine-tuning settings on the response process, respectively.

Model		BLEU	Entity Precision	Entity Recall	Entity F1
ChatGPT	base	24.1	67.5	69.9	68.7
	zeroCoT	19.0	57.8	85.3	68.9
	Uni-ETOD	26.2	70.9	69.9	70.4
	Uni-ETOD+KR(finetuned)	33.3	84.8	88.3	86.5
Gemini	base	37.3	73.3	81.0	76.9
	zeroCoT	38.9	75.9	83.0	79.3
	Uni-ETOD	43.0	77.3	89.3	82.9
	Uni-ETOD+KR(finetuned)	55.6	94.3	94.8	94.6
Llama3	base	10.2	63.7	70.5	66.9
	zeroCoT	19.1	65.4	77.7	71.0
	Uni-ETOD	15.2	76.0	82.4	79.0
	Uni-ETOD(fintuned)	56.0	94.5	94.1	94.3
Mistral	base	6.3	43.1	76.2	55.1
	zeroCoT	6.5	40.2	75.7	52.5
	Uni-ETOD	12.4	69.1	88.7	77.7
	Uni-ETOD(fintuned)	55.0	95.5	90.3	92.8
ChatGLM	base	8.1	39.0	73.7	51.0
	zeroCoT	6.7	36.6	75.2	49.2
	Uni-ETOD	21.8	76.2	87.7	81.6
	Uni-ETOD(fintuned)	43.9	91.8	86.6	89.1

Table 4: Experimental results of the response process on FakeRest dataset.

As shown in Table4, Uni-ETOD significantly improves the quality of responses compared to the baseline method. Uni-ETOD effectively mitigates the low precision problem of the retrieval process, which is reflected in the quality of responses. Uni-ETOD can better utilize the retrieval results with high recall and precision to generate more comprehensive and high confidence responses. In particular, the fine-tuned Uni-ETOD (Llama3) achieves a BLEU as high as 56 percent and Entity F1 as high as 94.3, outperforming ChatGPT and Gemini with zero-shot setting.

The experiments demonstrate that Uni-ETOD consistently enhance the performance of LLMs in the response process of Fully ETOD. We argue that Uni-ETOD can effectively stimulate LLMs' multi-hop reasoning in Fully ETOD. By guiding LLMs to gradually understand users' needs, Uni-ETOD can provide users with higher-quality and more credible responses.

5.4 Ablation Study

First, we evaluate the role of each step in Uni-ETOD in the retrieval process. Due to computational resource constraints, we performed ablation experiments on the fine-tuned Llama3 on FakeRest.

Model	Precision@10	Recall@10	F1@10
Uni-ETOD	96.0	96.7	96.4
w/o KR	17.0	98.9	29.0
w/o UR	97.8	91.5	94.6
w/o KR & UR	10.4	60.4	17.7

Table 5: Ablation study of the retrieval process on FakeRest dataset.

The table 5 demonstrates the impact of Retrieval Based on User Needs (RU), and Knowledge Refinement (KR) on the Fully ETOD retrieval process. Our RU step contains fine-tuning of the embedding model. For a fair comparison, we fine-tuned dialogue history as the query for "w/o RU" and "w/o RU & UR", and Uni-ETOD still achieved better results. The results show that the RU step can effectively improve the recall of the original paradigm retrieval process. Additionally, KR can effectively alleviate the low precision problem in the retrieval process.

We evaluate the role of each step in Uni-ETOD in the response process. We perform ablation experiments on fine-tuned Llama3, ChatGLM, and Mistral on FakeRest.

Model		BLEU	Entity Precision	Entity Recall	Entity F1
Llama3	Uni-ETOD	56.0	94.5	94.1	94.3
	w/o KR	40.7	85.0	88.9	85.0
	w/o UR	51.5	94.2	90.7	92.4
	w/o KR & UR	34.9	82.6	80.4	81.5
Mistral	Uni-ETOD	55.0	95.5	90.3	92.8
	w/o KR	43.9	85.6	88.9	87.2
	w/o UR	52.0	95.1	88.5	91.7
	w/o KR & UR	36.4	85.5	79.4	82.3
ChatGLM	Uni-ETOD	43.9	91.8	86.6	89.1
	w/o KR	29.8	55.2	83.8	66.5
	w/o UR	39.7	91.5	82.8	86.9
	w/o KR & UR	27.0	57.4	76.0	65.4

Table 6: Ablation study of the response process on FakeRest dataset.

The table 6 demonstrates the effects of Retrieval Based on User Needs (RU) and Knowledge Refinement (KR) on the Fully ETOD response process. The enhancement brought in the retrieval process is also shown in the response process. The results show that both RU and KR can effectively mitigate the low precision problem, improving the credibility, and overall quality of the responses.

6 Conclusion

In this paper, we aim to address the problems of low precision and poor interpretability in Fully ETOD. We propose a user-need-driven CoT framework (Uni-ETOD), which allows LLMs to gradually understand user needs and generate high-quality responses through multi-step reasoning. Experimental results demonstrate that Uni-ETOD effectively alleviates the low precision problem and offers users a more explanatory retrieval process and more reliable responses. Furthermore, we present a technique for automatically generating dialog data based on large-scale knowledge bases and constructing FakeRest, a dialogue dataset for restaurant scenarios.

Limitations

There are two limitations of this paper that deserve a deeper examination. First, we have not explored the fine-tuning methods and sampling techniques for embedding models in depth. Second, the method in this paper can be fine-tuned to adapt to various scenarios by automatically generating dialog data. However, our method is still not an autonomous learning method to adapt the system to new scenarios through user interaction.

References

AI@Meta. 2024. Llama 3 model card.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Mihail Eric and Christopher D Manning. 2017. Key-value retrieval networks for task-oriented dialogue. *arXiv preprint arXiv:1705.05414*.

Michael Heck, Nurul Lubis, Benjamin Matthias Ruppik, Renato Vukovic, Shutong Feng, Christian Geishauser, Hsien chin Lin, Carel van Niekerk, and Milica Gavsi'c. 2023. Chatgpt for zero-shot dialogue state tracking: A solution or an opportunity? *ArXiv*, abs/2306.01386.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. *Advances in Neural Information Processing Systems*, 33:20179–20191.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Guanhuan Huang, Xiaojun Quan, and Qifan Wang. 2022. Autoregressive entity generation for end-to-end task-oriented dialog. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 323–332, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Vojtěch Hudeček and Ondřej Dušek. 2023. Are large language models all you need for task-oriented dialogue? In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 216–228.

Léo Jacqmin, Lina M. Rojas Barahona, and Benoit Favre. 2022. "do you follow me?": A survey of recent approaches in dialogue state tracking. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 336–350, Edinburgh, UK. Association for Computational Linguistics.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Andrea Madotto, Samuel Cahyawijaya, Genta Indra Winata, Yan Xu, Zihan Liu, Zhaojiang Lin, and Pascale Fung. 2020. Learning knowledge bases with parameters for task-oriented dialogue systems. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2372–2394, Online. Association for Computational Linguistics.

Wenbo Pan, Qiguang Chen, Xiao Xu, Wanxiang Che, and Libo Qin. 2023. A preliminary evaluation of chatgpt for zero-shot dialogue understanding. *arXiv preprint arXiv:2304.04256*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics*.

Soham Parikh, Quaizar Vohra, Prashil Tumbade, and Mitul Tiwari. 2023. Exploring zero and few-shot techniques for intent classification. *arXiv preprint arXiv:2305.07157*.

Dongqi Pu and Vera Demberg. 2023. Chatgpt vs human-authored text: Insights into controllable text summarization and sentence style transfer. In *Annual Meeting of the Association for Computational Linguistics*.

Libo Qin, Wenbo Pan, Qiguang Chen, Lizi Liao, Zhou Yu, Yue Zhang, Wanxiang Che, and Min Li. 2023. End-to-end task-oriented dialogue: A survey of tasks, methods, and future directions. In *Proceedings of the*

2023 Conference on Empirical Methods in Natural Language Processing, pages 5925–5941, Singapore. Association for Computational Linguistics.

Libo Qin, Xiao Xu, Wanxiang Che, and Ting Liu. 2020. AGIF: An adaptive graph-interactive framework for joint multiple intent detection and slot filling. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1807–1816, Online. Association for Computational Linguistics.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. RocketQAv2: A joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2825–2835, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lina Maria Rojas-Barahona, Milica Gaić, Nikola Mrksic, Pei hao Su, Stefan Ultes, Tsung-Hsien Wen, Steve J. Young, and David Vandyke. 2016. A network-based end-to-end trainable task-oriented dialogue system. In *Conference of the European Chapter of the Association for Computational Linguistics*.

Md. Rony, Ricardo Usbeck, and Jens Lehmann. 2022. Dialokg: Knowledge-structure aware task-oriented dialogue generation. *ArXiv*, abs/2204.09149.

Weizhou Shen, Yingqi Gao, Canbin Huang, Fanqi Wan, Xiaojun Quan, and Wei Bi. 2023. Retrieval-generation alignment for end-to-end task-oriented dialogue system. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8261–8275, Singapore. Association for Computational Linguistics.

Devendra Singh, Siva Reddy, Will Hamilton, Chris Dyer, and Dani Yogatama. 2021. End-to-end training of multi-document reader and retriever for open-domain question answering. *Advances in Neural Information Processing Systems*, 34:25968–25981.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Xin Tian, Yingzhan Lin, Mengfei Song, Siqi Bao, Fan Wang, Huang He, Shuqi Sun, and Hua Wu. 2022. Q-tod: A query-driven task-oriented dialogue system. *arXiv preprint arXiv:2210.07564*.

Fanqi Wan, Weizhou Shen, Ke Yang, Xiaojun Quan, and Wei Bi. 2023. Multi-grained knowledge retrieval for end-to-end task-oriented dialog. In *Annual Meeting of the Association for Computational Linguistics*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Z. Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jianyun Nie, and Ji rong Wen. 2023. A survey of large language models. *ArXiv*, abs/2303.18223.

A Experimental Details of Constructing FakeRest Dataset

We utilize Prompt 1, 2, 3 to allow user LLMs and system LLMs to have multiple rounds of dialogues to build the dataset. Based on the user need path, we pass a user need as input to the user LLM, such as "find the moderate british restaurant in the center area.", and propose the need using Prompt 1. Based on the need path we can also query the entities from the knowledge base that match the need. If the query finds an entity that matches the need, the Prompt is used to reply to the user. If no entity is found that matches the need, then Prompt is used to apologize. We have the system LLM start the dialogue with "What can I do for you?", but we don't save this sentence in the dialogue dataset.

We used gemini-1.5-pro (Team et al., 2023) to generate the dialog dataset. To generate a more diverse set of responses, we use a temperature of 2.0. other hyperparameters use default values.

B Experimental Details of Fine-tuning Embedding Models

In the retrieval process, we are using bge-large-v1.5-en (Xiao et al., 2023) as the base model. We utilize the AdamW optimizer (Loshchilov and Hutter, 2017) and the linear learning rate scheduler with 0.1 warmup steps. We set the epoch to 2 and the learning rate to 2e-5. We utilize cosine similarity to compute the most relevant set of entities in the knowledge base.

Prompt 1 Prompt for user LLM

You are a user. Please respond to assistant based on the need of this round.
Dialogue Context:
Assistant: What can I do for you?
User: Can you recommend a good British restaurant in the centre of town?
Assistant: There are a few British restaurants in the centre of town, including Bistro Delights, Epicurean Emporium, and Tastebud Temptress.
Need: find the moderate british restaurant in the centre area.

Prompt 2 Prompt for system LLM

You are the assistant. Based on the 'Entities', please respond a sentence to the user.
Dialogue Context:
Assistant: What can I do for you?
User: Can you recommend a good British restaurant in the centre area?
Entities:
{'name': 'bistro delights', 'food': 'british', 'area': 'centre'},
{'name': 'epicurean emporium', 'food': 'british', 'area': 'centre'},
{'name': 'tastebud temptress', 'food': 'british', 'area': 'centre'}

Prompt 3 Prompt for system LLM without entities

You are the assistant. Please express sorry for not finding the restaurant meets the needs.
Dialogue Context:
Assistant: What can I do for you?
User: I'm looking for a European restaurant in the north area. Could you recommend one?
Assistant: I recommend either Spice Haven or Tropical Treats, both of which serve European cuisine in the north area.
User: Actually, I'm looking for something a little more upscale. Do you have any other suggestions for expensive European restaurants in the north area?