OPERA: Harmonizing Task-Oriented Dialogs and Information Seeking Experience

MIAORAN LI, Iowa State University, USA
BAOLIN PENG, Microsoft Research, USA
JIANFENG GAO, Microsoft Research, USA
ZHU ZHANG, University of Rhode Island, USA

Existing studies in conversational AI mostly treat task-oriented dialog (TOD) and question answering (QA) as separate tasks. Towards the goal of constructing a conversational agent that can complete user tasks and support information seeking, it is important to develop a system that can handle both TOD and QA with access to various external knowledge sources. In this work, we propose a new task, Open-Book TOD (OB-TOD), which combines TOD with QA and expands the external knowledge sources to include both explicit sources (e.g., the web) and implicit sources (e.g., pre-trained language models). We create a new dataset OB-MultiWOZ, where we enrich TOD sessions with QA-like information-seeking experience grounded on external knowledge. We propose a unified model OPERA (Open-book End-to-end Task-oriented Dialog) which can appropriately access explicit and implicit external knowledge to tackle the OB-TOD task. Experimental results show that OPERA outperforms closed-book baselines, highlighting the value of both types of knowledge.

CCS Concepts: • Information systems → World Wide Web; • Computing methodologies → Natural language processing; Machine learning.

Additional Key Words and Phrases: Web search, Task-oriented dialog systems, Language models

1 INTRODUCTION

Constructing conversational AI is a task full of challenges and has recently received extensive attention in the natural language processing (NLP) and information retrieval (IR) communities. Specifically, question answering (QA) systems and task-oriented dialog (TOD) systems are two important categories of conversational agents in practice [14].

QA systems aim at answering natural language questions by leveraging knowledge from large-scale data sources. Chen et al. [6] propose to use Wikipedia as an external knowledge source to tackle open-domain QA tasks and introduces a retriever-reader framework. The retriever, using a TF-IDF based approach, searches for relevant documents within a large corpus, while the reader extracts the answer from the retrieved passages. Subsequent research has focused on improving the performance of QA tasks through enhanced retrieval techniques [29, 30] and the expansion of external knowledge sources to include diverse sources of knowledge [7, 19, 57]. The increasing popularity of pre-trained language models [3, 48, 49] has also led to exploration of the use of implicit knowledge in these models for QA tasks [53, 61].

The dataset and code will be released to the community.

Authors’ addresses: Miaoran Li, limr@iastate.edu, Iowa State University, USA; Baolin Peng, Microsoft Research, USA, baolin.peng@microsoft.com; Jianfeng Gao, Microsoft Research, USA; jfgao@microsoft.com; Zhu Zhang, University of Rhode Island, USA, zhuzhang@uri.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
1559-1131/2023/9-ART $15.00
https://doi.org/10.1145/3623381

ACM Trans. Web
TOD systems, with different goals from QA systems, are expected to complete various user tasks (e.g., booking tickets and finding restaurants) in a conversational manner [4]. While current TOD systems are able to complete basic tasks, they may be unable to provide additional information not contained in pre-defined databases when asked follow-up questions that require such information. In an effort to enrich TODs with external knowledge, Kim et al. [27] propose a more challenging TOD setting in which models must detect knowledge-seeking turns, select relevant knowledge snippets from collected frequently asked questions (FAQs), and generate responses. More recent work has focused on the use of extended belief tracking to better manage structured knowledge in pre-defined databases and unstructured knowledge from collected snippets [16, 17].

Despite numerous efforts to create efficient and high-quality QA systems and TOD systems, single-task systems are still far from being true intelligent conversational agents. This is because users’ information needs in single-task problem settings are well-defined, and the system is not required to be aware of which task it should perform. In contrast, human conversations often involve multiple information-seeking tasks; therefore, conversational agents should be able to identify which task they are handling and take appropriate action. Additionally, current research on incorporating external knowledge to support information seeking in TODs has focused on limited knowledge sources, such as collected FAQ snippets [27], and does not incorporate up-to-date knowledge.

Figure 1 illustrates an example dialog involving both TOD and open-domain QA tasks. The first two turns are similar to the traditional TOD task, where information can be retrieved from pre-defined databases. The last two turns, however, necessitate the utilization of external knowledge sources to provide the required information. To manage such situations effectively, a system must be able to identify such information-seeking turns and consult the appropriate knowledge sources. Nehring et al. [40] combine TOD and open-domain QA using a modular framework that functions as a personal assistant and provides answers using Wikipedia. However, this approach may not be adequate for handling real-world dialogs that demand information that cannot be explicitly found. For example, in the last turn of the dialog in Figure 1, the user inquires about the limitations on WiFi usage on a train. The system predicts that this information cannot be found in the pre-defined databases and attempts to retrieve an answer from the Web but is unsuccessful. Traditional methods often produce meaningless responses, such as “I do not know,” when information from databases and the Web is insufficient to meet the demands of human-like information seeking.

To address research gaps in fused tasks and insufficient knowledge sources, we propose a more challenging and realistic dialog modeling task called Open-Book TOD (OB-TOD). This task not only combines TOD and QA tasks to better imitate human conversations with information-seeking behavior, but also generalizes the potential external knowledge sources to include the Web and pre-trained language models. We create a new dataset OB-MultiWOZ by leveraging existing datasets [4, 27]. Inspired by studies on implicit knowledge capacity of pre-trained language models [44, 53, 61], we develop a strategy for integrating implicit knowledge from these models with commonly used knowledge sources (e.g., databases, the Web) to better handle information-seeking turns in dialogs. We also propose an end-to-end system, based on previous work [43], that can handle QA turns that require external knowledge in the process of TOD. This system predicts a state based on the dialog history and uses this prediction for information retrieval, the results of which are then used to generate responses. Experimental results indicate that models with the ability to utilize both external implicit and explicit knowledge outperform models without access to external knowledge or only consulting explicit knowledge sources.

Specifically, we expect to make the following contributions:

- We formulate a general fused task, OB-TOD, that models both TOD and QA with access to explicit and implicit external knowledge sources in order to more closely mimic the information-seeking process in human-like conversations.
Fig. 1. Example of the defined task fusing TOD modeling and open-domain QA. The first two turns are similar to traditional TOD, while the last two turns involve open-domain QA with access to both explicit and implicit external knowledge sources. The information requested in the last turn cannot be found on the Web (an explicit knowledge source), and thus the use of implicit knowledge from pre-trained language models may be beneficial.

- Based on existing datasets, we construct OB-MultiWOZ dataset for OB-TOD. Dialog sessions in this dataset contain QA turns that require external information in the process of user task completion, and thus more closely resemble real-world conversations.
- We develop a unified model called OPERA with the ability to appropriately access explicit and implicit knowledge for the OB-TOD task. This model uses a predicted state to handle knowledge source selection and information retrieval, and is trained in an end-to-end manner to jointly optimize TOD and QA modeling. Experimental results show that OPERA has strong performance on the fused task.

2 RELATED WORK
2.1 Dialog Systems
Generally, dialog systems can be categorized into two classes based on whether it is aimed to accomplish some task proposed by a human user [37]. TOD systems are designed to complete certain tasks by interacting with users, while open-domain dialog (ODD) systems are expected to engage in social chats.

2.1.1 TOD Systems. There has been significant progress in the development of TOD systems in recent years, with a shift from modularized modeling towards end-to-end modeling approaches. End-to-end training allows for the joint optimization of the entire task, leading to improved overall performance for task completion [62]. Pre-trained language models have also been leveraged for end-to-end TOD modeling, with approaches such as modeling TOD as a single sequence prediction problem [23] and task-grounded pre-training over external data.
dialogue corpora [43, 55]. Zhang et al. [69] propose UBAR to model TOD on a dialog session level to better mimic the real-world scenarios. He et al. [22] tend to inject the knowledge of dialog policy explicitly into pre-training to further improve TOD systems.

In addition to training strategies, many works focus on expanding knowledge coverage to better handle user requests. Unstructured knowledge-grounded task-oriented conversational modeling [27] involves the use of external knowledge and requires models to decide whether or not to access this knowledge based on the dialog history. The task can be divided into three subtasks: knowledge-seeking turn detection, knowledge selection, and knowledge-grounded generation. Many previous works on this task focus on improving knowledge selection strategy [21, 25, 36, 58]. Extended belief states have also been introduced to handle TOD grounded in both structured and unstructured knowledge, allowing for both database querying and document retrieval [16, 17]. Some approaches have explored the use of multiple knowledge sources to improve the performance of knowledge-enhanced TOD systems, including the integration of knowledge from various local datasets [64–66]. Unlike previous approaches, our focus is on selecting appropriate knowledge resources rather than individual knowledge snippets. We extend the task of knowledge-seeking turn detection to include the decision of whether to consult external knowledge resources, as well as the selection of the appropriate (implicit or explicit) knowledge source to access.

2.1.2 ODD Systems. Early dialogue systems without explicit knowledge integration often produce less engaging and meaningful responses due to their lack of real-world grounding [24, 54, 60]. To improve the quality of responses, researchers have explored various methods for incorporating external knowledge into dialog systems. The methods include creating a knowledge-grounded conversation dataset covering a wide range of topics [18], utilizing a knowledge base comprising Wikipedia articles [12], and using the Web as a source of knowledge to generate responses with more accurate information [28]. Other approaches involve externalizing implicit commonsense knowledge through matching dialogs with a knowledge graph dataset and training models to generate both knowledge and responses given a dialog history [71]. In this work, we consider both the Web and pre-trained language models as external knowledge sources.

2.1.3 Dialog Systems for Fused Task. Researchers have started studying fused task settings that aim to more closely mimic human-level conversations by handling various types of dialogs, such as constructing conversational recommendation systems that can make appropriate recommendations [11, 20, 34, 70], or enhancing TOD systems with ODDs to make dialogs more natural and engaging [8, 9, 56, 67]. Few works enable TOD systems to answer questions with access to external knowledge, such as the combination of a TOD system and open-domain QA using a modular framework that can serve as a personal assistant and answer questions from Wikipedia [40]. This system has a module selection component that chooses between a module for open-domain QA and one for TOD completion based on the user’s utterance. However, independently formulating TOD and QA tasks and simply combining the components can prevent information sharing and lead to suboptimal performance. In this work, we propose a unified end-to-end model called OPERA that jointly models TOD and QA tasks. In addition, previous work has relied on either explicit or implicit knowledge for QA, while OPERA learns to automatically select the appropriate knowledge source for each task.

2.2 Open-domain Question Answering

Open-domain question answering (QA) is a task of finding answers to general-domain questions without specified context by searching a large set of documents, either locally or on the Web. Early approaches to this task typically use the retriever-reader framework, with the retriever component responsible for finding relevant passages from a large collection of documents and the reader component designed to find the answer within these retrieved passages [6]. Retrievers are initially based on techniques such as TF-IDF or BM25, while readers are implemented
using neural models. To overcome the recall ceiling of untrainable retrievers, later works start exploring the use of dense representations for the retriever component [26, 29, 30].

Some researchers have also attempted to improve QA performance by incorporating a wider range of external knowledge, including both open-domain and in-domain knowledge [41], as well as knowledge from different domains [5] or heterogeneous modalities such as text, images, and tables [7, 19, 33, 45, 57]. With the advent of large-scale pre-trained language models [3, 48, 49], some researchers have attempted to utilize implicit knowledge from these models to tackle QA tasks [53, 61] instead of focusing on improving retrieval efficiency and expanding knowledge coverage.

In this work, we consider both explicit and implicit knowledge sources for knowledge acquisition and propose a model that can consult these sources appropriately to respond to multi-domain questions in natural language. Our task setting is more complex and realistic compared to previous work that mainly focused on answering questions in restricted domains or choosing an option from a list of candidates, as our model is expected to learn to access explicit and implicit external knowledge appropriately to satisfy users’ information-seeking requirements.

2.3 Conversational Search and Conversational Question Answering

Conversational search and conversational question answering (ConvQA) are two subtasks of conversational information retrieval [15, 68], which is a special information retrieval task in the setting of multi-turn conversations in audio or text.

Conversational search [1, 2, 59] involves the seeking of information through conversation, which may involve multiple exchanges and the handling of conversation history in order to generate an appropriate query to retrieve relevant passages. The conversational query understanding task [51] is a subtask of conversational search that focuses on understanding the query within the context of the conversation. OR-QuAC [47] is a benchmark for conversational search that requires models to retrieve knowledge from a large collection in order to generate answers.

Previous work on ConvQA has mainly focused on two categories [14]: conversational machine comprehension (CMC) task [10, 50, 72] and sequential knowledge-base question answering (KB-QA). Our work is more related to the CMC task, as we only consider unstructured external knowledge. CMC task involves answering questions given a passage (or a set of passages) and the conversation history in the form of question-answer pairs. Most research in this area has used the CoQA [50] and QuAC [10] datasets, which involve conversations centered on the provided passages. Different from previous work on answering complex factual questions, [46] proposes using machine reading comprehension to generate contentful conversations, including chitchat and non-factual responses. Ren et al. [52] introduce a more challenging problem setting in which systems are required to answer a query given query history and a list of retrieved candidate passages from the search engine. The new task includes more realistic scenarios where users’ questions cannot be answered by a short text span extracted from the given passages.

Our work combines elements of conversational search and ConvQA, but differs in several ways. More specifically, (1) Our primary task is to complete task-oriented dialogs that may include the seeking of external information, while ConvQA works mainly focus on answering questions; (2) Instead of already knowing the current turn requires external knowledge, our task requires systems to predict whether it is a knowledge-seeking turn based on dialog history, which sets higher demands on models; (3) Besides widely used external knowledge resources, such as the Web or large knowledge base, we also consider pre-trained LMs with implicit knowledge as possible external knowledge sources; (4) Our task requires models to predict the knowledge source to consult and formulate an appropriate query based on dialog history.

Our work. Table 1 provides a high-level comparison of our work and previous research. There are not many works on the fused task setting of TOD and QA. We propose a new task setting that seamlessly fuses TOD and
Table 1. High-level summary of related work. “ConvSearch” is short for conversational search. “PLMs” corresponds to pre-trained language models. “w/ exp. kn.” (or “w/ imp. kn.”) represents QA with access to explicit (or implicit) knowledge. We consider large-scale corpus such as Wikipedia as a Web knowledge source.

| Related Work   | Task                        | Knowledge sources | Training method |
|---------------|-----------------------------|-------------------|-----------------|
|               |                             | Database | Web | PLMs |                     |
| Peng et al. [43] | TOD                         | ✓        | ✓   | ✓    | End-to-end           |
| Komeili et al. [28] | ODD                         | ✓        | ✓   | ✓    | End-to-end           |
| Sun et al. [56] | TOD + ODD                   | ✓        | ✓   | ✓    | End-to-end           |
| Chiu et al. [9] | TOD + ODD                   | ✓        | ✓   | ✓    | Modular              |
| Nehring et al. [40] | TOD + QA w/ exp. kn. | ✓        | ✓   | ✓    | Modular              |
| Pan et al. [41] | QA w/ exp. kn.              | ✓        | ✓   | ✓    | End-to-end           |
| Wang et al. [61] | QA w/ imp. kn.              | ✓        | ✓   | ✓    | End-to-end           |
| Qu et al. [47] | ConvSearch + ConvQA         | ✓        | ✓   | ✓    | End-to-end           |
| Ren et al. [52] | ConvQA                      | ✓        | ✓   | ✓    | End-to-end           |
| This work     | TOD + QA                    | ✓        | ✓   | ✓    | End-to-end           |

QA tasks and generalizes the possible knowledge sources to include databases, the Web, and pre-trained language models. We create a new dataset and design an end-to-end model for our task.

3 TASK FORMULATION

In a traditional TOD modeling task [4], the system is assumed to access a back-end database to obtain information required by the user. Kim et al. [27] extend TOD task setting to integrate unstructured external knowledge from collected FAQs. In this work, we propose a more challenging task, open-book TOD (OB-TOD), which extends previous TOD tasks in two aspects: (1) We fuse TOD and QA tasks and require models able to handle QA turns occurring in TODs that ask for external knowledge; (2) We generalize possible external knowledge sources, especially explicit knowledge sources (e.g., the Web) and implicit knowledge sources (e.g., pre-trained language models), to enhance standard TOD systems.

Figure 2 shows the overview of the OB-TOD task. For each turn in the dialog, the model is required to first predict state $s$ based on dialog history $h$ up to the current turn. The state indicates the appropriate knowledge source to access for knowledge acquisition. The model then performs grounded generation to generate response $r$.

The full task consists of three processes: state prediction, knowledge retrieval, and knowledge-grounded response generation. We use off-the-shelf models for knowledge retrieval and do not consider it a subtask. In the implementation, the retrieval function can be database lookup, Bing search engine for explicit external knowledge, or GPT-3 for implicit external knowledge. We break the full task into two subtasks: state prediction and knowledge-grounded response generation. In $t$-th dialog turn, given dialog history $h = \{u_{t-k}, r_{t-k}, \ldots, u_t\}$, where $u_t$ and $r_t$ represent user utterance and system response in the $i$-th turn, respectively, and $k$ is the history window size, the model predicts state $s$ informing which knowledge source will be accessed and the query used

Please note that the use of “open-book” here underscores expanding the TOD agent’s capability with access to external knowledge, in contrast to the traditional “closed-book” TOD architecture with no expectation of external information seeking. It is different from “open book QA” [39], where the emphasis is on multi-step reasoning over commonsense knowledge in a QA setting.

ACM Trans. Web
Fig. 2. Overview of our task. We split OB-TOD into two subtasks: (1) state prediction and (2) grounded response generation to obtain knowledge from the selected knowledge source, which is formulated as

\[ s = f_s(h), \]

where \( f_s \) is a sequence-to-sequence model that takes dialog history \( h \) as input.

If external knowledge is not required, the model obtains a database state from the pre-defined database. Database state in plain text is regarded as knowledge \( k \) for response generation. Otherwise, the model retrieves either explicit or implicit external knowledge for response generation. It is assumed that the process of knowledge retrieval is deterministic given the state \( s \), as the database lookup process is deterministic and external knowledge sources tend to be unchanged. Finally, the model generates response \( r \) based on dialog history \( h \), predicted state \( s \) and retrieved knowledge \( k \):

\[ r = f_r(h, s, k), \]

where \( f_r \) is a sequence-to-sequence model with dialog history \( h \), predicted state \( s \) and retrieved knowledge \( k \) as input.

4 OB-MULTIWOZ DATASET

Since exiting TOD or QA datasets do not embody the tight integrative nature of the OB-TOD task, we build a new dataset OB-MultiWOZ to facilitate our work and future efforts by the community.

4.1 Dataset Construction

The OB-MultiWOZ dataset is created by adding QA-style information-seeking turns to TOD sessions. These turns are based on data from the original MultiWOZ 2.1 and DSTC9 Track1 datasets. The DSTC9 Track1 dataset is created by crowd workers who were asked to identify a suitable position for inserting a knowledge-seeking turn, write a user utterance for that position, and generate a system response based on relevant knowledge snippets collected from FAQs. However, in the collection of OB-MultiWOZ dataset, workers were expected to retrieve knowledge from the Web and generate responses based on selected knowledge. The OB-MultiWOZ dataset extends the information-seeking turns in the DSTC9 Track1 dataset from closed-domain to open-domain setting. It replaces closed-domain knowledge candidates from FAQ webpages with open-domain knowledge retrieved from the Web and further includes implicit knowledge snippets generated by PLMs. To create this dataset, we first collected search queries, explicit knowledge, and responses on Amazon Mechanical Turk, and then used GPT-3 to augment the implicit knowledge.
Fig. 3. The pipeline of data collection. Given dialog context and user utterance, a worker was asked to (a) generate an appropriate query for the user question, (b) select relevant knowledge snippets from retrieval results using the generated query, and (c) write a response based on the selection in (b) or mark the question as unanswerable. After collecting data from crowd workers, we use GPT-3 to generate implicit knowledge for unanswerable questions.

4.1.1 Crowdsourcing. The crowdsourcing workflow is shown in Figure 3. Given a TOD with an inserted user utterance seeking external information, crowd workers were asked to write down a search query for the user question and use it to retrieve relevant information from a search engine (Figure 3a). After receiving retrieval results, workers needed to identify whether the results were useful. If workers found valuable passages in retrieval results, they were required to select those passages (Figure 3b) and then use that knowledge to write a response to the user’s utterance (Figure 3c). If the search results were not useful, or if it would be difficult to answer the user’s question based on the retrieved passages, the workers were instructed to mark the question as unanswerable (Figure 3c). In such cases, system utterances from the original dataset are used as responses to the unanswerable question.

4.1.2 Answerable and Unanswerable Questions. A user question is considered answerable if the crowd workers can find useful information in the search results using the generated query. The workers generated the response to an answerable question based on the selected knowledge. If the workers determined that it was difficult to answer the user’s question based on the retrieved passages, the question was marked as unanswerable, and the response was recorded as “not answerable” (Figure 3c). In such cases, system utterances from the original dataset are used as responses to the unanswerable question.

An example of an unanswerable question is shown in Figure 4. The dialog history and current user utterance (in blue) are displayed on the left, and the top-ranked retrieved passages are listed on the right. In this example, the crowd worker used the search query “credit cards acceptance in Alimentum restaurant” (in the blue box) to retrieve information from the Bing search engine. However, none of the retrieved passages were found to be useful, so the worker marked the user’s utterance as an unanswerable question.
4.1.3 Implicit Knowledge Augmentation. Unanswerable questions require information outside the scope of the pre-defined database and are found unable to be answered using explicit knowledge. Therefore, we believe that implicit knowledge may provide valuable context and common-sense knowledge that can help generate more informative responses to these questions. After collecting search queries, selected knowledge (explicit knowledge), user question types (answerable or unanswerable), and responses, we augmented the dataset by adding implicit knowledge for unanswerable QA turns using GPT-3 [3]. GPT-3 has the ability to learn in context, meaning it can quickly adapt to new tasks with only a few examples in the inference phase without fine-tuning. We considered two methods for generating implicit knowledge using GPT-3 (Figure 5).

The first method for generating implicit knowledge is to utilize GPT-3 as a policy model. This is accomplished by providing the model with in-context learning examples in the form of sampled dialogs that are concatenated with the current dialog history, and using the generated system response as implicit knowledge. Through learning from these in-context examples, GPT-3 is expected to generate a response that is appropriate to the current user utterance. The in-context examples are presented in the form of <user utterance>

\n<system utterance>

\n<system utterance>

\n, where \n is used as in-dialog separator and \n is used to separate different sample dialogs. The upper part of Figure 5 shows the example of implicit knowledge generation by using GPT-3 as a policy model based on dialog history for the unanswerable question shown in Figure 4.

The other method for generating implicit knowledge is to use GPT-3 as a knowledge base. This is achieved by providing the model with the concatenation of in-context examples and an annotated query. The in-context example comprises of search queries and selected knowledge extracted from sampled answerable questions, presented in the form <query>

\n<knowledge>

\n. By training on this prompt format, it is expected that GPT-3 will learn to generate a knowledge snippet that is similar to the explicit knowledge of answerable questions based on a given search query. The lower part of Figure 5 shows an example of implicit knowledge generation by using GPT-3 as a knowledge base based on annotated search query for the unanswerable question shown in Figure 4.
Fig. 5. Example of two methods to obtain implicit knowledge using GPT-3. The upper part (in blue) shows an example of using GPT-3 as a policy model to obtain implicit knowledge. The lower part (in green) gives an example of using GPT-3 as a knowledge base for implicit knowledge acquisition.

4.2 Statistics
Table 2 includes statistics of the collected dataset. We randomly sampled 1202 dialogs with 1643 inserted user utterances from the DSTC9 Track1 dataset for data collection. These dialogs were split into 379 for training, 41 for validation, and 782 for testing. Out of the inserted user questions, 1312 were annotated as answerable and 331 were marked as unanswerable. On average, there are 1-2 QA turns in a dialog.

5 MODEL
We propose a unified model OPERA (Open-book End-to-end Task-oriented Dialog) for the OB-TOD task. OPERA is novel in two aspects: (1) it seamlessly incorporates information seeking into end-to-end TODs, and (2) it is equipped with a knowledge source selection mechanism to leverage explicit and implicit knowledge for task completion. The illustration of the proposed model is shown in Figure 6. OPERA first predicts a state, which indicates the knowledge source to consult and the query used to obtain knowledge, based on the dialog history. Off-the-shelf models are then used to acquire knowledge based on the state prediction. Finally, OPERA performs grounded generation to generate a response.

5.1 State Prediction
State $s$ tracks a user’s goal throughout a dialog. In particular, a state $s$ is in the form $ks: q$, where $ks$ represents the knowledge source to consult, and $q$ stands for query used to acquire knowledge from the predicted source.

---

3 We used a subset of DSTC9 dataset as we focus on the few-shot training setting which is often the case in real-world situations.
4 Please see footnote 2 regarding the "open-book" terminology.
Table 2. Statistics of OB-MultiWOZ. The dataset consists of training, validation, and test sets. Each split can be further broken into Answerable and Unanswerable subsets based on the type of inserted questions. There is no overlap between the Answerable subset and the Unanswerable subset. Total Question turns represents the total number of answerable QA turns in the Answerable subset and the total number of unanswerable QA turns in the Unanswerable subset. Avg. QA turns is the average number of QA turns within a dialog. Avg. Query Len. is the average number of tokens in the queries. Avg. Knowledge Len. represents the average length of the selected knowledge snippets. Avg. Answer Len. is the average length of responses for QA turns. $K_B$ denotes the mean length of implicit knowledge generated by using GPT-3 as knowledge base, while $P_L$ represents the mean length of knowledge generated by using GPT-3 as policy model.

| Split            | Train   | Validation | Test    |
|------------------|---------|------------|---------|
| Total Dialog     | 323     | 56         | 35      | 6       | 523     | 259     |
| Total TOD turns  | 2737    | 348        | 285     | 41      | 3824    | 1888    |
| Total QA turns   | 641     | 56         | 78      | 6       | 593     | 269     |
| Avg. QA turns    | 1.98    | 1.00       | 2.23    | 1.00    | 1.13    | 1.04    |
| Avg. Query Len.  | 5.04    | 6.39       | 5.06    | 8.33    | 5.35    | 5.97    |
| Avg. Knowledge Len. | 58.96  | 51.68$K_B^{1/2}$ | 59.71 | $44.50K_B^{1/2}$ | 62.05 | $^{1/2}$ |
| Avg. Answer Len. | 11.51   | 20.36      | 10.38   | 19.00   | 13.59   | 20.34   |

Fig. 6. Overall architecture of the OPERA model

In this work, we consider three possible knowledge sources: pre-defined database, explicit external knowledge source, and implicit external knowledge source. The query for accessing a pre-defined database is a belief state in traditional TOD modeling, and the query for accessing explicit or implicit external knowledge sources is in the form of a search query.

$$s = \begin{cases} 
\text{Database: belief state,} & \text{model predicts to access database,} \\
\text{Explicit: search query,} & \text{model predicts to access explicit knowledge source,} \\
\text{Implicit: search query,} & \text{model predicts to access implicit knowledge source.}
\end{cases}$$  \( (3) \)

Examples of possible states are shown in Figure 7. Suppose the model predicts that external knowledge is not needed to respond current turn (Figure 7a). In such a case, the predicted state indicates that the selected knowledge
source is a pre-defined database, followed by a belief state, for example, Database: restaurant pricerange = expensive food = Chinese area = north. If the model predicts that the user’s question is seeking external information, either the explicit knowledge source (Figure 7b) or the implicit knowledge source (Figure 7c) can be accessed by the predicted search query, for example, Explicit: cancel taxi booking extra charge or Implicit: credit cards acceptance in Alimentum restaurant.

Given dialog history \( h = \{u_{t-k}, r_{t-k}, ..., u_t\} \), where \( u_i \) and \( r_i \) represent user utterance and system response at turn \( i \), respectively, and \( k \) is the history window size, the training objective of state prediction can be formulated as

\[
L_S = \log p(s \mid h) = \sum_{i=1}^{N_t} \log p_{\theta}(s_i \mid s_{<i}, h),
\]

where \( \theta \) represents trainable parameters in the model, \( N_t \) is the target length of predicted state sequence, and \( s_{<i} \) denotes tokens before index \( i \). In the implementation, we add a task-specific prefix \([49]\) to input to specify which task the model should perform. The input of state prediction is in the form State Prediction: <dialog history>.

5.2 Knowledge Acquisition

Off-the-shelf models are adopted for knowledge acquisition. In the implementation, the retrieval function can be database lookup, a search engine for explicit external knowledge, or a large-scale pre-trained language model for implicit external knowledge.

5.2.1 Database. If OPERA predicts to query the pre-defined database (Figure 7a) based on predicted state \( s \), predicted belief state is used to query the database \([4]\). A belief state is a list of triplets in the form (domain, slot_name, value) recording values for slots in a particular domain. A database state that contains records satisfying the conditions of the belief state is returned and used as knowledge for response generation.

5.2.2 Explicit Knowledge Source. We use Bing Search API as an explicit knowledge source. It can be easily generalized to other explicit knowledge sources such as Wikidump. In the example in Figure 7b, the predicted state \( s \) is Explicit: cancel taxi booking extra charge. Bing API is triggered, and retrieval results based on query cancel taxi booking extra charge are returned.

5.2.3 Implicit Knowledge Source. In our implementation, we regard GPT-3 \([3]\) as an implicit knowledge source (Figure 7c), which can be replaced by other large-scale pre-trained language models. GPT-3 is proven to have the ability of in-context learning, which means it can be quickly adapted to new tasks with only a few examples in the inference phrase without fine-tuning. As shown in Figure 8, we propose two types of approaches to obtaining implicit knowledge from GPT-3.

Using GPT-3 as a policy model. The intuition is to ask GPT-3 to respond to user utterances directly and utilize the generated response as implicit knowledge. Generating response to user questions requires GPT-3 to complete the closed-book QA task \([53]\), which involves utilizing implicit knowledge obtained in the pre-training process. The input to GPT-3 is comprised of in-context examples and dialog history. Two example passages are provided for in-context learning. The example is in the form of <user utterance>\n<system utterance>\n... <system utterance>\n\n. In-context examples and dialog history are concatenated as input prompt to GPT-3 (upper part of Figure 8), and the generated system utterance is used as implicit knowledge.

Using GPT-3 as a knowledge base. The other method to acquire implicit knowledge is to query GPT-3 as the way we request Bing API. The input prompt to GPT-3 consists of in-context examples and the predicted query. The in-context example is compromised of search queries and selected knowledge of sampled answerable QA turns,
(a) Example of using pre-defined database

(b) Example of using external explicit knowledge source

(c) Example of using external implicit knowledge source. Two methods are used to obtain implicit knowledge.

Fig. 7. Examples of the proposed model using different knowledge sources
Fig. 8. Illustration of two methods to access implicit knowledge source. We use GPT-3 either as a policy model or knowledge base to acquire implicit knowledge.

in the form: \(<query>\|n<knowledge>\|n\|n\). The query in the predicted state is appended to in-context learning examples, and the concatenated string is used as an input prompt to GPT-3 (lower part of Figure 8).

5.3 Grounded Response Generation

System response \(r = \{r_1, r_2, ..., r_{N_r}\}\) with length \(N_r\) is generated grounded on dialog history \(h\), predicted state \(s\) and retrieved knowledge \(k\). The objective is defined as

\[
L_R = \log p(r \mid h, s, k) = \sum_{i=1}^{N_r} \log p_{\theta}(r_i \mid r_{<i}, h, s, k). \tag{5}
\]

With task-specific prefix for grounded generation, the input is in the form Response Generation: \(<dialog history> \| <knowledge> \| <selected knowledge>\), where \(<knowledge>\) is a special token indicating the beginning of retrieved knowledge.

5.4 Training Objective of Full Task

Each training example is represented as:

\(x = (h, s, k, r)\), \tag{6}

where \(h = \{u_{-k}, r_{-k}, ..., u_t\}\) is the dialog history consisting of user and system utterance in the last \(k - 1\) turns and current user utterance, \(s\) is the state including knowledge source to retrieve and the corresponding query, \(k\) is the retrieved knowledge, and \(r\) is the (delexicalized) dialog response. The knowledge \(k\) can be a database state in plain text, a passage in Bing retrieval results, or implicit knowledge from GPT-3.

Combining learning objectives (4) and (5) of subtasks, the joint objective of full task is

\[
\mathcal{L}_\theta(\mathcal{D}) = \sum_{i=1}^{n} (L_S(x_i) + L_R(x_i)), \tag{7}
\]
OPERA: Harmonizing Task-Oriented Dialogs and Information Seeking Experience • 15

where \( D = \{x_i\}_{i=1}^n \) is training dataset consisting of \( n \) training examples. We use a Transformer to parameterize both state prediction and grounded response generation process, and parameters in \( \theta \) are updated by maximizing the joint objective.

6 EXPERIMENTS

We perform end-to-end evaluations of the proposed model to answer two questions: (1) How does the proposed model handle OB-TOD task? (2) How does the proposed model leverage explicit and implicit knowledge to respond to information-seeking questions when completing TODs?

6.1 Experimental Setup

6.1.1 Evaluation Metrics and Datasets. We consider three evaluation settings: (1) standard TOD completion as described in \([4, 13]\), (2) QA task, and (3) full task involving TOD and QA.

In the TOD evaluation setting, following previous works \([4, 13]\) we measure whether the model provides an appropriate entity (\( \text{Inform} \)), e.g., restaurant location or price range, and answers all the request attributes (\( \text{Success} \)), e.g., phone number or postcode, in dialog-level. We use BLEU \([42]\) to measure how fluent the generated responses are compared to human-annotated answers. A combined score \([38]\) \( \text{Combine} = (\text{Inform} + \text{Success}) \times 0.5 + \text{BLEU} \) is computed as an overall measure of generation quality.

In the QA setting, we report BLEU score of generated answers for QA turns. We measure whether the model predicts correct external knowledge sources to consult (\( \text{Accuracy} \)) and generates appropriate search queries (\( \text{Query F1} \)), as well as its potential for success in QA turns (\( \text{Success Rate} \)).

Query F1 is the F1 score between a predicted search query and annotated query to measure token-level overlap, which can be computed by

\[
\text{precision} = \frac{\#\text{common tokens}}{\#\text{tokens in predicted query}}, \\
\text{recall} = \frac{\#\text{common tokens}}{\#\text{tokens in golden query}}, \\
\text{Query F1} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}},
\]

where \( \#\text{common tokens} \) is the number of tokens that occur in both prediction and golden query. \( \text{Accuracy} \) is calculate by

\[
\#\text{detect}_{\text{explicit}} = \sum_{i=1}^{N} \mathbb{1}(k_{spred} = \text{explicit} \& k_{sgold} = \text{explicit}), \\
\#\text{detect}_{\text{implicit}} = \sum_{i=1}^{N} \mathbb{1}(k_{spred} = \text{implicit} \& k_{sgold} = \text{implicit}), \\
\text{Accuracy} = \frac{\#\text{detect}_{\text{explicit}} + \#\text{detect}_{\text{implicit}}}{N} \times 100,
\]

where \( \mathbb{1}(\cdot) \) is the indicator function, \( N \) is the total number of dialogs containing inserted QA turns in the test set, \( k_{spred} \) and \( k_{sgold} \) represent predicted and annotated knowledge sources, respectively. \( \#\text{detect}_{\text{explicit}} \) (or \( \#\text{detect}_{\text{implicit}} \)) is the number of correctly predicted states indicating to use explicit (or implicit) external knowledge source. Success Rate is a measure of the model’s potential for success in QA turns. Achieving a
reasonable Query F1 score within the dialog is a prerequisite for success in the QA task. Success Rate is computed as follows

\[
\frac{\sum_{i=1}^{N} \left( \sum_{j=1}^{n_i} \text{Query-F1}(q_{pred}, q_{gold}) \geq 20 \right)}{N},
\]

where \(n_i\) is the number of inserted QA turns in \(i\)-th dialog. \(q_{pred}\) and \(q_{gold}\) represent predicted query and annotated search query, respectively.

In the full task setting, we report BLEU, Inform, Success and Combine. BLEU score is computed on all TOD and QA responses. The computation of Inform and Success differs from that in the TOD setting and is limited to successful dialogs. A dialog is considered successful in the full task if it succeeds in both TOD modeling and QA tasks. In the implementation, the potential success of the QA task, as determined by the average Query F1 score being no less than 20, is used as an indicator and Inform and Success are only calculated on dialogs whose average Query F1 score of QA turns is at least 20.

6.1.2 Models. We train five models in an end-to-end manner and evaluate them in experiments. Training settings for models are summarized in Table 3.

**T5 (T)** is the closed-book baseline, which can only handle TOD (T) modeling and is unable to access external knowledge. It is trained on TOD turns in OB-MultiWOZ and not exposed to the QA task in the training process.

**T5 (T + Q)** is the model with both TOD (T) and QA (Q) skills but not exposed to external knowledge sources during training and still in the closed-book setting. It only makes use of the history of QA turns during training. Search queries and external knowledge for QA turns are hidden. Compared to T5 (T), it has “memorized” some knowledge in training and can utilize “memorized” information to answer questions.

**T5 (T + Q) w/ EK** is the model with TOD (T) modeling skill and able to handle QA (Q) with access to explicit external knowledge (w/ EK). It is trained on full OB-MultiWOZ. We utilize dialog histories, search queries, and selected knowledge for answerable QA turns and only use dialog histories for unanswerable QA turns. Queries and external knowledge for unanswerable questions are not provided during training. Therefore the model is expected to utilize explicit knowledge appropriately and answer questions requiring implicit information using memorized knowledge.

**OPERA-GPT3PM** is the proposed model that can handle TOD and QA tasks. It can access explicit knowledge from the Web and implicit knowledge using GPT-3 as a policy model (GPT3PM). It is trained on full OB-MultiWOZ. We use dialog histories, search queries, and external knowledge for all QA turns in training. The augmented implicit knowledge is GPT-3 generated responses based on dialog histories for unanswerable questions. This model is anticipated to have the capability to consult explicit or implicit knowledge as needed in inference. When the model predicts to consult an implicit knowledge source, it prompts GPT-3 by dialog history and uses the generated response as knowledge.

**OPERA-GPT3KB** is the proposed model that can handle TOD and QA tasks. It can acquire explicit knowledge from the Web and implicit knowledge using GPT-3 as a knowledge base (GPT3KB). It is trained on full OB-MultiWOZ with a similar setting to OPERA-GPT3PM. The only difference is that the augmented implicit knowledge is GPT-3 generated knowledge snippets based on search queries. The model prompts GPT-3 using the predicted query.

6.1.3 Implementation Details. The implementation of models is based on Huggingface Pytorch Transformer [63]. T5-base model. Training examples were truncated with (or padded to) length of 512. To make sure input strings

\[\text{We analyzed the relationship between Query F1 and the Combined score of the full task and found the performance of the models decreases significantly as the threshold of Query F1 increases from a minimum of 10 to a minimum of 20, especially for closed-book baselines T5 (T) and T5 (T + Q). This suggests that a Query F1 score of 20 is a decent bar of performance for pre-trained models on QA tasks. Therefore, we use Query F1 ≥ 20 as the criteria to compute Success Rate of QA task.}\]
Table 3. Training settings of models. Mark ✓ represents data available during training, and mark X denotes the information not provided. T5 (T) represents the model only with TOD (T) modeling skill while T5 (T + Q) denotes the model with both TOD (T) and QA (Q) skills. T5 (T + Q) w/ EK represents the model with TOD and QA skills and the ability to access explicit knowledge (w/ EK). OPERA-GPT3PM (OPERA-GPT3KB) indicates that the model uses GPT-3 as a policy model (knowledge base) to obtain implicit knowledge.

| Model                        | TOD turns | QA turns | External knowledge source           | Explicit | Implicit |
|------------------------------|-----------|----------|-------------------------------------|----------|----------|
| T5 (T)                       | ✓         | ✗        |                                     |          |          |
| T5 (T + Q)                   | ✓         | ✓        |                                     |          |          |
| T5 (T + Q) w/ EK             | ✓         | ✓        | Bing                                |          |          |
| OPERA-GPT3PM                 | ✓         | ✓        | Bing                                | GPT-3 as policy model |          |
| OPERA-GPT3KB                 | ✓         | ✓        | Bing                                | GPT-3 as knowledge base |          |

contain both dialog history and retrieved knowledge, we truncated dialog history on the left with a maximum length of 256. The history window size was set to 2 and therefore dialog history \( h = \{ u_{t-2}, r_{t-3}, u_{t-1}, r_{t-1}, u_t \} \) contains five utterances. We used AdamW optimizer [35] with constant learning rate 0.001. Models were trained with a mini-batch of 6 on 4 Nvidia Tesla K80. The training is stopped early when no decrease in validation loss can be observed to avoid overfitting or up to 20 epochs. We conducted five runs of experiments for each setting using different random seeds.

6.2 Main Results

Table 4. End-to-End evaluation of the full task on OB-MultiWOZ

| Model                        | BLEU     | Full Task Evaluation | Combined |
|------------------------------|----------|-----------------------|----------|
|                              | Success  | Inform                |          |
| T5 (T)                       | 13.37(0.28) | 6.09(1.90)           | 7.93(1.93) | 20.38(2.16) |
| T5 (T + Q)                   | 14.34(0.42) | 7.08(1.35)           | 9.33(1.53) | 22.55(1.63) |
| T5 (T + Q) w/ EK             | 13.73(0.53) | 31.46(3.66)          | 44.58(2.33) | 51.75(3.24) |
| OPERA-GPT3PM                 | 14.04(0.32) | 37.75(3.01)          | 52.46(4.84) | 59.14(3.93) |
| OPERA-GPT3KB                 | 14.32(0.65) | **44.64**(3.75)      | **58.49**(4.90) | **65.88**(4.02) |

The results of the main experiment on OB-MultiWOZ are presented in Table 4. It is evident that the OPERA models outperform the other models in the OB-TOD task, as indicated by the Combined score. The T5 (T + Q) w/ EK model, which has access to explicit Web knowledge, is able to complete the full task, highlighting the importance of incorporating external knowledge for the OB-TOD task. The OPERA-GPT3KB model, which leverages both explicit and implicit knowledge, achieves the highest Combined score, demonstrating the effectiveness of the proposed model and the importance of incorporating various forms of knowledge.

To gain a better understanding of the performance variations, we present the evaluation results for individual tasks in Table 5. The results for the TOD modeling task can be found in the left column of the table. It can be observed that there is little difference in TOD completion among the models, with the largest difference in the Combined score being less than 7 points. However, the OPERA-GPT3KB model exceeds the T5 (T) model by over 45 points in the full task. This suggests that the ability to effectively leverage external knowledge for the QA task makes OPERA distinguished in the full task. The right section of Table 5 presents the evaluation results for the
Table 5. End-to-End evaluation of single tasks on OB-MultiWOZ.

| Model                  | TOD Evaluation | QA Evaluation |
|------------------------|----------------|---------------|
|                         | BLEU           | Success       | Inform | Combined | Accuracy | Success Rate | Query F1 | BLEU         |
| T5 (T)                 | 14.59 (0.26)   | 45.32 (4.52)  | 60.02 (3.85) | 67.26 (4.35) | 0.00 (0.00) | 13.43 (2.65) | 9.02 (0.95) | 2.60 (0.24) |
| T5 (T + Q)             | **14.85 (0.47)** | 47.04 (1.09)  | 60.84 (1.75) | 68.79 (1.52) | 0.00 (0.00) | 15.45 (3.07) | 9.50 (0.90) | 6.00 (0.33) |
| T5 (T + Q) w/ EK       | 14.14 (0.63)   | 44.50 (4.36)  | 62.12 (2.96) | 67.45 (3.90) | 68.79 (0.00) | 71.84 (0.72) | 48.28 (0.33) | 5.86 (0.24) |
| OPERA-GPT3PM           | 14.41 (0.35)   | 43.16 (4.67)  | 59.60 (6.91) | 65.79 (5.76) | 89.49 (4.20) | 87.34 (3.01) | 57.91 (1.86) | 5.56 (0.16) |
| OPERA-GPT3KB           | 14.74 (0.76)   | **49.76 (2.94)** | **65.02 (4.07)** | **72.13 (3.05)** | **91.21 (3.85)** | **88.67 (3.09)** | **58.97 (1.84)** | **6.22 (0.39)** |

QA task. While the T5 (T + Q) model can generate more natural responses to questions compared to the T5 (T) model, both lack the ability to select appropriate external knowledge sources and are therefore unable to complete the QA task. The T5 (T + Q) w/ EK model, which has access to external knowledge sources, performs significantly better than these models. However, there are still some differences in performance when compared to the OPERA models. The OPERA-GPT3KB model exhibits superior performance in selecting appropriate knowledge sources and predicting precise search queries. The performance gap between the T5 (T + Q) w/ EK and OPERA models indicates that incorporating both explicit and implicit external knowledge sources can be beneficial for the QA task.

6.3 Evaluation on Answerable Questions

Table 6. End-to-End evaluation results on Answerable questions to demonstrate the importance of explicit external knowledge.

| Model                  | Full Task Evaluation | Combined |
|------------------------|----------------------|----------|
|                         | BLEU                 | Success  | Inform | Combined |
| T5 (T)                 | 13.38 (0.26)         | 4.71 (1.65) | 6.19 (1.83) | 18.83 (1.91) |
| T5 (T + Q)             | **14.73 (0.32)**     | 6.39 (1.81) | 8.53 (2.17) | 22.19 (2.19) |
| T5 (T + Q) w/ EK       | 14.13 (0.55)         | 42.54 (4.46) | 60.86 (2.60) | 65.85 (3.79) |
| OPERA-GPT3PM           | 14.34 (0.35)         | 41.79 (5.09) | 58.22 (7.16) | 64.34 (6.11) |
| OPERA-GPT3KB           | 14.65 (0.62)         | **48.39 (2.74)** | **63.70 (4.05)** | **70.70 (2.88)** |

The evaluation results for the full task on Answerable are presented in Table 6. Similar to what we have observed in Table 4, the T5 (T + Q) w/ EK and OPERA models are able to complete the full task on answerable questions by leveraging explicit knowledge from the Web. Without access to external knowledge sources, the T5 (T) and T5 (T + Q) models struggle to achieve good performance in the full task on answerable questions. This variation in performance highlights the importance of explicit knowledge for the OB-TOD task. It is also worth noting that there is a slight improvement in the performance of the T5 (T + Q) model compared to the T5 (T) model, which suggests that the knowledge memorized by the T5 (T + Q) model during training can provide some benefit for the full task.

The evaluation results for individual tasks on the Answerable subset are shown in Table 7. All models are able to complete the TOD task, with little difference in performance. However, when it comes to the QA task on answerable questions, the T5 (T) and T5 (T + Q) models are not able to perform well without access to external knowledge. The T5 (T + Q) model, which has knowledge memorized from the training process, is able to generate more natural responses than the T5 (T) model, but is still outperformed by models that have the
Table 7. End-to-End evaluation results of single tasks on Answerable questions to demonstrate the importance of explicit external knowledge for the QA task seeking external information.

| Model | TOD Evaluation | QA Evaluation | Full Task Evaluation |
|-------|----------------|--------------|----------------------|
|       | BLEU | Success | Inform | Combined | BLEU | Success Rate | Query F1 | BLEU |
| T5 (T) | 14.59(0.24) | 46.32(3.77) | 61.84(3.35) | 68.67(3.69) | 0.00(0.00) | 9.48(2.14) | 7.39(0.70) | 2.06(0.22) |
| T5 (T + Q) | **14.92(0.38)** | **47.62(1.61)** | **61.64(2.17)** | **69.55(1.98)** | 0.00(0.00) | **13.84(3.72)** | **8.64(1.11)** | **7.66(0.58)** |
| T5 (T + Q) w/ EK | 14.25(0.60) | 43.80(4.88) | 62.46(2.96) | 67.38(4.19) | **100.00(0.00)** | **97.17(0.28)** | **64.67(0.34)** | **7.76(0.35)** |
| OPERA-GPT3PM | 14.48(0.33) | 43.24(5.04) | 60.14(7.14) | 64.16(7.06) | **100.00(0.00)** | **97.13(0.25)** | **64.31(0.48)** | **8.20(0.42)** |
| OPERA-GPT3KB | **14.84(0.70)** | **50.14(3.08)** | **65.78(4.17)** | **72.80(3.12)** | **100.00(0.00)** | **97.05(0.25)** | **64.16(0.48)** | **8.20(0.42)** |

ability to access explicit external knowledge. The T5 (T + Q) w/ EK and OPERA-GPT3KB models, which are able to acquire explicit knowledge from the Web, perform better in terms of selecting appropriate knowledge sources and generating natural responses, suggesting that leveraging external knowledge is crucial for success in the QA task on the Answerable subset. The strong performance of the T5 (T + Q) w/ EK model in handling answerable questions further emphasizes the importance of leveraging external knowledge for the QA task. The OPERA-GPT3KB model obtains better performance in generating natural responses, implying that incorporating implicit knowledge can benefit the ability to utilize explicit knowledge.

6.4 Evaluation on Unanswerable Questions

After determining that explicit external knowledge is necessary for effectively answering answerable questions, we then investigate whether the ability to access explicit knowledge alone is sufficient for achieving success in the Question Answering task.

Table 8. End-to-End evaluation results on Unanswerable questions to demonstrate the importance of implicit external knowledge.

| Model | BLEU | Full Task Evaluation | QA Evaluation | Full Task Evaluation |
|-------|------|----------------------|--------------|----------------------|
|       |      | Success | Inform | Combined | BLEU | Success Rate | Query F1 | BLEU |
| T5 (T) | 13.34(0.48) | 8.88(2.41) | 11.43(2.35) | 23.50(2.76) |
| T5 (T + Q) | 13.59(0.65) | 8.50(0.55) | 10.97(1.32) | 23.32(0.67) |
| T5 (T + Q) w/ EK | 12.89(0.58) | 9.11(2.06) | 11.66(1.90) | 23.27(2.28) |
| OPERA-GPT3PM | 13.40(0.25) | 29.64(2.38) | 40.86(2.65) | 48.65(2.62) |
| OPERA-GPT3KB | **13.60(0.80)** | **37.13(6.19)** | **47.95(8.14)** | **56.14(7.03)** |

The evaluation results of the OB-TOD task on the Unanswerable subset are presented in Table 8. The OPERA models, which have access to both explicit and implicit knowledge, outperform the other models by more than 32 points in the Combined score. It is observed that the T5 (T + Q) w/ EK model performs well in answering answerable questions by leveraging explicit knowledge from the Web. However, its performance is found to be inadequate when it comes to handling unanswerable questions, thus indicating that explicit knowledge is not effective for answering unanswerable questions.

The results of the evaluation on individual tasks are shown in Table 9. The OPERA-GPT3KB model performs slightly better in TOD modeling, but the differences are not significant. The discrepancies become more apparent in the QA task, where the OPERA models, which have access to the implicit knowledge source as needed, are more likely to succeed in more QA turns. On the other hand, the other three models, which lack the ability to
Table 9. End-to-End evaluation results of single tasks on unanswerable questions to demonstrate the importance of implicit external knowledge for the QA task.

| Model               | BLEU   | TOD Evaluation | QA Evaluation |
|---------------------|--------|----------------|---------------|
|                     |        | Accuracy   | Success | Inform | Combined | Success Rate | Query F1 | BLEU |
| T5 (T)              | 14.58(0.47) | 43.30(6.08) | 56.36(5.19) | 64.41(5.84) | 0.00(0.00) | 21.39(4.16) | 12.59(1.62) | 3.28(0.45) |
| T5 (T + Q)          | 14.70(0.68) | 45.80(1.64) | 59.22(1.85) | 67.21(1.30) | 0.00(0.00) | 18.69(2.90) | 11.40(0.97) | 3.39(0.68) |
| T5 (T + Q) w/ EK    | 13.91(0.72) | 45.90(3.34) | 61.28(3.12) | 67.50(3.52) | 0.00(0.00) | 20.70(2.26) | 12.15(0.58) | 2.87(0.44) |
| OPERA-GPT3PM        | 14.25(0.42) | 43.00(4.30) | 58.52(6.58) | 65.01(5.45) | 66.32(13.47) | 67.57(9.32) | 44.14(6.49) | 2.00(0.34) |
| OPERA-GPT3KB        | 14.54(0.93) | 49.04(2.73) | 63.48(3.97) | 70.80(3.02) | 71.82(12.36) | 71.74(9.63) | 47.21(6.58) | 3.20(0.76) |

consult the implicit knowledge source, are unable to provide users with useful information. The T5 (T + Q) model with memorized knowledge from the training process is able to generate more natural responses. The significant overall lead of the OPERA models demonstrates the effectiveness of the strategy to incorporate both implicit and explicit external knowledge.

6.5 Is implicit external knowledge enough for QA?

The previous analysis suggests that implicit external knowledge is crucial for the success of the QA and full tasks. Next, we investigate whether implicit knowledge alone can eliminate the need for explicit knowledge.

Table 10 shows the evaluation results of the GPT-3 and OPERA models on all QA turns. We use GPT-3 as a control model because it has been demonstrated to have strong abilities in various NLP tasks, including question answering [3] and knowledge retrieval [31]. We also use GPT-3 generation as implicit knowledge in the training procedure, so it is considered to contain rich implicit knowledge.

When compared to the OPERA models, which are able to utilize both explicit and implicit knowledge, GPT-3 performs better in responding to unanswerable questions but falls behind significantly in answerable questions. This result indicates the importance of explicit knowledge in the QA task. Our strategy of combining explicit and implicit knowledge does benefit the QA task.

6.6 Evaluation of State Prediction

Table 10. Evaluation results on all QA turns to demonstrate the necessity to incorporate explicit and implicit knowledge. BLEU scores are reported in the table.

| Model               | Answerable | Unanswerable | Overall |
|---------------------|------------|--------------|---------|
| GPT-3               | 2.20       | 3.73         | 2.80    |
| OPERA-GPT3PM        | 7.86       | 2.00         | 5.56    |
| OPERA-GPT3KB        | 8.20       | 3.20         | 6.22    |

Table 11. Evaluation of state prediction on OB-MultiWOZ. "Accuracy" is the accuracy of knowledge source classification. "Slot F1" is the micro-averaged F1 score between the predicted belief state and the ground truth over all slots [32].

| Model               | TOD Accuracy | Slot F1 | Answerable Accuracy | Query F1 | Unanswerable Accuracy | Query F1 |
|---------------------|--------------|---------|---------------------|----------|-----------------------|----------|
| T5 (T)              | 100.00(0.00) | 34.72(1.36) | 0.00(0.00) | 7.39(0.70) | 0.00(0.00) | 12.59(1.62) |
| T5 (T + Q)          | 100.00(0.00) | 35.43(0.60) | 0.00(0.00) | 8.64(1.11) | 0.00(0.00) | 11.40(0.97) |
| T5 (T + Q) w/ EK    | 100.00(0.00) | 33.64(1.59) | 100.00(0.00) | 64.67(0.34) | 0.00(0.00) | 12.15(0.58) |
| OPERA-GPT3PM        | 99.96(0.04) | 33.63(0.73) | 100.00(0.00) | 64.16(0.34) | 66.32(13.47) | 44.14(6.49) |
| OPERA-GPT3KB        | 99.92(0.10) | 33.76(1.71) | 100.00(0.00) | 64.31(0.48) | 71.82(12.36) | 47.21(6.58) |
We evaluate the performance of the models in the state prediction subtask by assessing the accuracy of knowledge source prediction (Accuracy) and the appropriateness of the predicted query. For TOD turns, we adopt the evaluation methodology from previous work [32] and use the Slot F1 score to indicate the correctness of the predicted belief state. For QA turns, we report the Query F1 score as a measure of the appropriateness of the query.

The results of the state prediction on the full dataset are shown in Table 11. It can be observed that all models are able to accurately select the appropriate knowledge source for TOD turns, as well as generate appropriate belief states. The T5 (T + Q) w/ EK and OPERA models also perform well in predicting the use of the explicit knowledge source, and in generating relatively accurate search queries for answerable questions. For unanswerable questions, only the OPERA models are able to utilize the implicit knowledge source, and generate better search queries. While the OPERA models significantly outperform the baselines, they do not always select the correct knowledge source. This may be due to the difficulty in accurately classifying unanswerable questions, which may lack distinctive patterns or features that distinguish them from answerable questions and TOD turns. Additionally, the size of the training data for unanswerable questions is significantly smaller compared to that for answerable questions, which may limit the models’ ability to generalize to this type of question. Overall, there is still room for improvement in the models’ ability to predict the appropriate knowledge source for unanswerable questions.

6.7 Human Evaluation
We conducted a turn-level pairwise comparison for human evaluation to assess OPERA’s performance. We randomly selected 300 responses for each model and 100 utterances for each type (i.e., TOD, answerable, and unanswerable turns). We hired workers with a lifetime HIT acceptance rate of greater than 95% and presented them with the dialog history and two responses from different models. The workers were asked to rate the responses on three dimensions using a 5-point Likert scale: 1) Usefulness measures how well the response provides the expected information; 2) Humanness measures the fluency and coherence of the response with the dialog context; Safety measures whether the response is socially safe (i.e., does not contain toxic, biased, or misleading content). We compared three pairs of models: 1) T5 (T + Q) and T5 (T + Q) w/ EK as competitive baselines, 2) OPERA-GPT3PM and OPERA-GPT3KB as variants of the proposed model, and 3) T5 (T + Q) w/ EK and OPERA-GPT3KB as a competitive baseline and the proposed model, respectively.

The results of human evaluation are reported in Table 12. The 5-point Likert is converted into a Win/Tie/Loss scale. The left column contains the comparison between T5 (T + Q) and T5 (T + Q) w/ EK. We report the percentage that T5 (T + Q) w/ EK wins, ties with, and loses to T5 (T + Q). As expected, the T5 (T + Q) w/ EK model outperforms the T5 (T + Q) model on all three metrics for answerable responses, due to its access to external explicit knowledge. However, contrary to the results of the automatic evaluation, the T5 (T + Q) w/ EK model is also able to generate more human-like responses than the T5 (T + Q) model, indicating a potential gap between automatic metrics and human evaluation. The percentage that OPERA-GPT3KB wins, ties with, and loses to OPERA-GPT3PM are shown in the middle column. Consistent with the results of the automatic evaluation, the OPERA-GPT3PM model has a stronger ability to generate useful and fluent responses. This suggests that using GPT-3 as a knowledge base is a more effective strategy. The right column shows the results of the comparison of T5 (T + Q) w/ EK and OPERA-GPT3KB, including the probability that OPERA-GPT3KB wins, ties with, and loses to T5 (T + Q) w/ EK. Overall, there is no obvious difference in the informativeness of responses. Our proposed model generates responses with significantly better fluency, indicating that our strategy to utilize both explicit and implicit knowledge is beneficial to designing more human-like dialog systems. However, OPERA-GPT3KB model seems to be less safe, possibly due to the lack of quality control for the generated implicit knowledge. There is still room for improvement in terms of generating high-quality implicit knowledge and training systems with better grounded generation abilities.
Table 12. Human evaluation results. In the left section, "Win" (or "Loss") stands for the percentage that T5 (T + Q) w/ EK wins (or loses). In the middle and right sections, "Win" (or "Loss") represents the percentage that OPERA-GPT3KB wins (or loses).

| Setting     | Metric     | T5 (T + Q) w/ EK vs. T5 (T + Q) | OPERA-GPT3KB vs. OPERA-GPT3PM | OPERA-GPT3KB vs. T5 (T + Q) w/ EK |
|-------------|------------|---------------------------------|---------------------------------|-----------------------------------|
|             |            | Win | Tie | Loss | Win | Tie | Loss | Win | Tie | Loss |
| Answerable  | Usefulness | 32.0 | 42.0 | 26.0 | 25.0 | 44.0 | 31.0 | 30.0 | 36.0 | 34.0 |
|             | Humanness  | 40.0 | 43.0 | 17.0 | 30.0 | 48.0 | 22.0 | 33.0 | 50.0 | 17.0 |
|             | Safety     | 27.0 | 55.0 | 18.0 | 18.0 | 62.0 | 20.0 | 19.0 | 60.0 | 21.0 |
| Unanswerable| Usefulness | 35.0 | 33.0 | 32.0 | 34.0 | 39.0 | 27.0 | 34.0 | 35.0 | 31.0 |
|             | Humanness  | 43.0 | 37.0 | 20.0 | 39.0 | 37.0 | 24.0 | 39.0 | 32.0 | 29.0 |
|             | Safety     | 26.0 | 51.0 | 23.0 | 24.0 | 53.0 | 23.0 | 19.0 | 57.0 | 24.0 |
| TOD         | Usefulness | 32.0 | 43.0 | 25.0 | 32.0 | 41.0 | 27.0 | 30.0 | 42.0 | 28.0 |
|             | Humanness  | 37.0 | 47.0 | 16.0 | 32.0 | 47.0 | 21.0 | 39.0 | 39.0 | 22.0 |
|             | Safety     | 25.0 | 58.0 | 17.0 | 25.0 | 54.0 | 21.0 | 21.0 | 57.0 | 22.0 |
| Overall     | Usefulness | 33.0 | 39.3 | 27.7 | 30.3 | 41.3 | 28.3 | 31.3 | 37.7 | 31.0 |
|             | Humanness  | 40.0 | 42.3 | 17.7 | 33.7 | 44.0 | 22.3 | 37.0 | 40.3 | 22.7 |
|             | Safety     | 26.0 | 54.7 | 19.3 | 22.3 | 56.3 | 21.3 | 19.7 | 58.0 | 22.3 |

6.8 Case Study

We show example dialog segments and corresponding output results of models (s, r) in Table 13. Without access to external knowledge sources, T5 (T + Q) predicts the knowledge source to be the pre-defined database by mistake in both examples and fails to provide expected information to the user. In the first example, T5 (T + Q) w/ EK and OPERA-GPT3KB can acquire knowledge from the Web and respond to the user with accurate information. In the second example, T5 (T + Q) w/ EK fails to consult implicit knowledge sources and gives incorrect information to the user. In contrast, OPERA-GPT3KB succeeds in providing correct information to users. However, we can notice that OPERA cannot provide as many details as humans. In the second example, the response of OPERA-GPT3KB only suggests that the taxi booking can be canceled, while the human response also includes restrictions on cancellation. This indicates that there is still room to improve OPERA to generate correct and detailed responses.

7 CONCLUSION AND FUTURE WORK

In this work, we define a new task OB-TOD that combines task-oriented dialog and information seeking via external knowledge (e.g., the Web and language models). We construct a new dataset OB-MultiWOZ with manually composed questions, search queries, and knowledge grounded responses. To tackle this task, we propose OPERA, a unified model that seamlessly incorporates QA capabilities seeking external information into an end-to-end task-oriented dialog agent. Experimental results on OB-MultiWOZ indicate that by combining implicit and explicit knowledge, OPERA is able to handle the fused task and outperform closed-book baselines. We believe that the newly proposed task OB-TOD may represent a significant step towards building human-like conversational AI agents, and OB-MultiWOZ can help facilitate research in this direction.
Table 13. Comparison of state prediction and response generation results when models occur inserted QA turns in OB-MultiWOZ. “Human” denotes ground truth results annotated by workers.

**Example 1**

|   |   |
|---|---|
| **U:** | I need a train leaving on Friday. |
| **R:** | Where are you departing from and what is your destination? |
| **U:** | The train should depart from Cambridge and should go to Ely. |
| **R:** | Is there a particular time you’d like to leave? |
| **U:** | My itinerary isn’t confirmed yet, so what is the cancellation policy for the train? |
| **T5 (T + Q)** | s : Dataset: train destination = ely ; day = friday ; departure = cambridge |
|   | r : There are [value_count] trains that meet your criteria. |
| **T5 (T + Q) w/ EK** | s : Explicit: cancellation policy for train |
|   | r : You can cancel your train ticket booking by calling them or cancel your booking by going to the app where you booked it to cancel the booking. |
| **OPERA-GPT3KB** | s : Explicit: cancellation policy for train |
|   | r : You can change or cancel your train ticket booking before the train departs. |
| **Human** | s : Explicit: cambridge train cancellation policy |
|   | r : Don’t worry, your ticket will be refunded. Any other questions? |

**Example 2**

|   |   |
|---|---|
| **U:** | I want to book a taxi to commute between the two places. |
| **R:** | To which direction will you be going? |
| **U:** | Going from El Shaddai to the Cow Pizza Kitchen and Bar, and I need to arrive in time for my reservation at the restaurant. |
| **R:** | Okay I will look into this for you and be back in a moment. |
| **U:** | Will I be able to cancel my taxi booking if my plans change later on? |
| **T5 (T + Q)** | s : Dataset: taxi destination = the cow pizza kitchen and bar ; departure = el shaddai |
|   | r : Yes, I have booked you a taxi. The contact number is [taxi_phone]. |
| **T5 (T + Q) w/ EK** | s : Dataset: taxi destination = the cow pizza kitchen and bar ; departure = el shaddai |
|   | r : I am sorry, but I am unable to cancel your booking. Would you like to try a different taxi booking? |
| **OPERA-GPT3KB** | s : Implicit: can I cancel my taxi booking later on |
|   | r : Yes, I will cancel your taxi booking if you change your mind later on. |
| **Human** | s : Implicit: el shaddai taxi cancel booking |
|   | r : You can cancel 24 hours in advance. After that there is a 10 gbp fee. Does this help you? |

For future work, we believe it is important to continue improving the accuracy and factualness of implicit knowledge. Additionally, it will be valuable to explore evaluation metrics that consider both humanness and the factual accuracy of responses. While our work focuses on adding QA capabilities to task-oriented dialog agents,
human information-seeking needs are diverse in real-world scenarios. Therefore, it would be useful to develop a more comprehensive task involving multiple skills, such as recommendation and personalization, and use it to benchmark the progress of human-like conversational AI agents.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers who critically review and provide insightful feedback on this research paper. Their constructive comments and suggestions significantly contributed to enhancing the clarity of our work.

REFERENCES

[1] Avishek Anand, Lawrence Cavedon, Matthias Hagen, Hideo Joho, Mark Sanderson, and Benno Stein. 2021. Dagstuhl Seminar 19461 on Conversational Search: Seminar Goals and Working Group Outcomes. SIGIR Forum 54, 1. Article 3 (Feb 2021), 11 pages. https://doi.org/10.1145/3451964.3451967

[2] Leif Aziz, Mateusz Dubiel, Martin Halvey, and Jeffery Dalton. 2018. Conceptualizing agent-human interactions during the conversational search process. In The second international workshop on conversational approaches to information retrieval.

[3] 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 (2020), 1877–1901.

[4] Pawel Budzianowski, Tsung-Hsien Tseng, Bo-Hsiang Tseng, Ihigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Brussels, Belgium, 5016–5026. https://doi.org/10.18653/v1/D18-1547

[5] Cen Chen, Chengyu Wang, Menghui Qiu, Dehong Gao, Linbo Jin, and Wang Li. 2021. Cross-Domain Knowledge Distillation for Retrieval-Based Question Answering Systems. In Proceedings of the Web Conference 2021 (Ljubljana, Slovenia) (WWW ’21). Association for Computing Machinery, New York, NY, USA, 2613–2623. https://doi.org/10.1145/3442381.3449814

[6] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to Answer Open-Domain Questions. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, 1870–1879. https://doi.org/10.18653/v1/P17-1171

[7] Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Yang Wang, and William W. Cohen. 2021. Open Question Answering over Tables and Text. In International Conference on Learning Representations.

[8] Zhiyu Chen, Bing Liu, Seungwhan Moon, Chinmudra Sankar, Paul Crook, and William Yang Wang. 2022. KETOD: Knowledge-Enriched Task-Oriented Dialogue. arXiv preprint arXiv:2205.05589 (2022).

[9] Suu Chiu, Maolin Li, Yen-Ting Lin, and Yun-Nung Chen. 2022. SalesBot: Transitioning from Chit-Chat to Task-Oriented Dialogues. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Dublin, Ireland, 6143–6158. https://aclanthology.org/2022.acl-long.425

[10] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question Answering in Context. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Brussels, Belgium, 2174–2184. https://doi.org/10.18653/v1/D18-1241

[11] Yang Deng, Wenzuan Zhang, Weiwen Xu, Wengiang Lei, Tat-Seng Chua, and Wai Lam. 2022. A Uniﬁed Multi-Task Learning Framework for Multi-Goal Conversational Recommender Systems. ACM Trans. Inf. Syst. (Nov 2022). https://doi.org/10.1145/3570640

[12] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia: Knowledge-powered Conversational Agents. In Proceedings of the International Conference on Learning Representations (ICLR).

[13] Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tür. 2020. MultiWOZ 2.1: A Consolidated Multi-Domain Dialogue Dataset with State Corrections and State Tracking Baselines. In Proceedings of the 12th Language Resources and Evaluation Conference. European Language Resources Association, Marseille, France, 422–428. https://aclanthology.org/2020.lrec-1.53

[14] Jianfeng Gao, Michel Galley, and Lihong Li. 2018. Neural Approaches to Conversational AI. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts. Association for Computational Linguistics, Melbourne, Australia, 2–7. https://doi.org/10.18653/v1/P18-5003

[15] Jianfeng Gao, Chenyan Xiong, Paul Bennett, and Nick Craswell. 2022. Neural Approaches to Conversational Information Retrieval. arXiv preprint arXiv:2201.05176 (2022).

[16] Silin Gao, Ryuichi Takanobu, Antoine Bosselut, and Minlie Huang. 2022. End-to-End Task-Oriented Dialog Modeling with Semi-Structured Knowledge Management. IEEE/ACM Transactions on Audio, Speech, and Language Processing (2022).
[57] Alon Talmor, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannanee Hajishirzi, and Jonathan Berant. 2021. MultiModalQA: complex question answering over text, tables and images. In International Conference on Learning Representations.

[58] David Thulke, Nico Daheim, Christian Dugast, and Hermann Ney. 2021. Efficient Retrieval Augmented Generation from Unstructured Knowledge for Task-Oriented Dialog. In Workshop at DSTC9, AAAI.

[59] Johanne R. Trippas, Damiano Spina, Falk Scholer, Ahmed Hassan Awadallah, Peter Bailey, Paul N. Bennett, Ryen W. White, Jonathan Liono, Yongli Ren, Flora D. Salim, and Mark Sanderson. 2019. Learning About Work Tasks to Inform Intelligent Assistant Design. In Proceedings of the 2019 Conference on Human Information Interaction and Retrieval (Glasgow, Scotland UK) (CHIIR ’19). Association for Computing Machinery, New York, NY, USA, 5–14. https://doi.org/10.1145/3295750.3298934

[60] Oriol Vinyals and Quoc Le. 2015. A neural conversational model. arXiv preprint arXiv:1506.05869 (2015).

[61] Cunxiang Wang, Pai Liu, and Yue Zhang. 2021. Can Generative Pre-trained Language Models Serve As Knowledge Bases for Closed-book QA?. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Association for Computational Linguistics, Online, 3241–3251. https://doi.org/10.18653/v1/2021.acl-long.251

[62] Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A Network-based End-to-End Trainable Task-oriented Dialogue System. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers. Association for Computational Linguistics, Valencia, Spain, 438–449. https://aclanthology.org/E17-1042

[63] Hamed Zamani, Johanne R. Trippas, Jef Dalton, and Filip Radlinski. 2022. Conversational Information Seeking. arXiv preprint arXiv:2201.08808 (2022).

[64] Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020. Task-oriented dialog systems that consider multiple appropriate responses under the same context. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34. 9604–9611.

[65] Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, Xiaoke Wang, and Ji-Rong Wen. 2020. Towards Topic-Guided Conversational Recommender System. In Proceedings of the 28th International Conference on Computational Linguistics. 4128–4139.

[66] Pei Zhou, Behnam Hedayatnia, Karthik Gopalakrishnan, Seokhwan Kim, Jay Pujara, Xiang Ren, Yang Liu, and Dilek Hakkani-Tur. 2021. Think Before You Speak: Learning to Generate Implicit Knowledge for Response Generation by Self-Talk. In Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI. Association for Computational Linguistics, Online, 251–253. https://doi.org/10.18653/v1/2021.nlpcvai-1.23

[67] Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2018. Sdnet: Contextualized attention-based deep network for conversational question answering. arXiv preprint arXiv:1812.03593 (2018).