End-to-End Task-Oriented Dialog Modeling with Semi-Structured Knowledge Management

Silin Gao, Ryuichi Takanobu, Minlie Huang†
CoAI Group, DCST, IAI, BNRIST, Tsinghua University, Beijing, China
gsl16@tsinghua.org.cn, gxly19@mails.tsinghua.edu.cn, aihuang@tsinghua.edu.cn

***Abstract*—Current task-oriented dialog (TOD) systems mostly manage structured knowledge (e.g. databases and tables) to guide the goal-oriented conversations. However, they fall short of handling dialogs which also involve unstructured knowledge (e.g. reviews and documents). In this paper, we formulate a task of modeling TOD grounded on a fusion of structured and unstructured knowledge. To address this task, we propose a TOD system with semi-structured knowledge management, SeKnow, which extends the belief state to manage knowledge with both structured and unstructured contents. Furthermore, we introduce two implementations of SeKnow based on a non-pretrained sequence-to-sequence model and a pretrained language model, respectively. Both implementations use the end-to-end manner to jointly optimize dialog modeling grounded on structured and unstructured knowledge. We conduct experiments on the modified version of MultiWOZ 2.1 dataset, where dialogs are processed to involve semi-structured knowledge. Experimental results show that SeKnow has strong performances in both end-to-end dialog and intermediate knowledge management, compared to existing TOD systems and their extensions with pipeline knowledge management schemes.**

***Index Terms*—task-oriented dialog, semi-structured knowledge management, end-to-end modeling**

I. Introduction

RECENT Task-Oriented Dialog (TOD) systems [2]–[8] have achieved promising performance on accomplishing user goals. Most systems typically query *structured knowledge* such as tables and databases based on the user goals, and use the query results showing matched entities to guide the generation of system responses, as shown in the first dialog turn in Fig. 1.

However, real-world task-oriented conversations also often involve *unstructured knowledge* [9] related to the user's entities of interest, such as reviews and regulation documents. For example, as the second dialog turn in Fig. 1 shows, the user asks about customers' favorite food at a matched restaurant *Pizza Hut*, which involves the customer reviews of this entity. Current TOD systems fall short of handling such dialog turns since they cannot utilize relevant unstructured knowledge. This deficiency may interrupt the dialog process, causing difficulties in tracking user goals and generating system responses.

In this paper, we consider incorporating more various forms of domain knowledge into the TOD systems. We define a task of modeling TOD which involves knowledge with both structured and unstructured contents, as shown in the semi-structured knowledge base in Fig. 1. In each dialog turn, the system needs to track the user goals associated with structured knowledge as triples and use them to query the knowledge base. The query results (i.e. the matched entities) are then used to generate the system response. Besides, the system also needs to retrieve the unstructured contents (i.e. the documents) of knowledge base according to user goals and select relevant references (if existed) for generating the response.

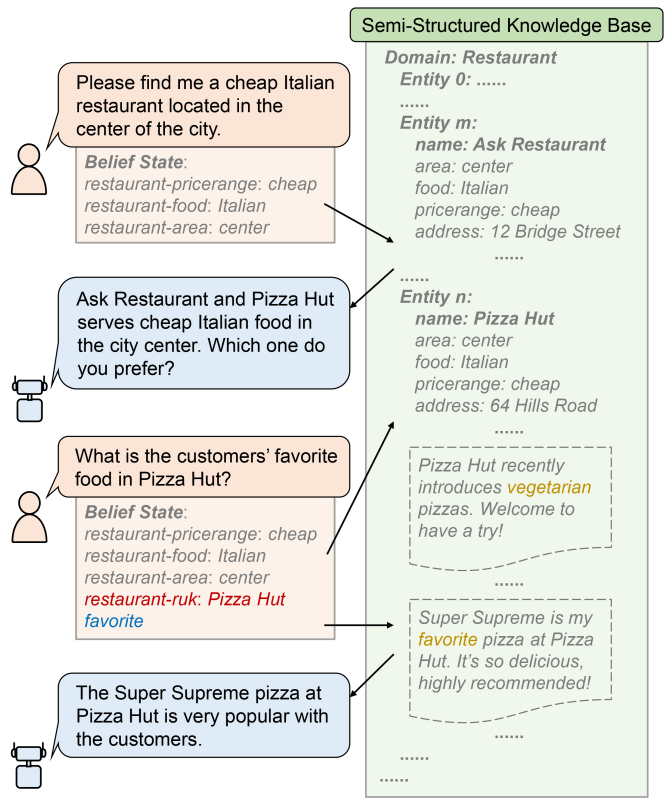

Fig. 1. Illustration of task-oriented dialog modeling with semi-structured knowledge management. Words in red and blue illustrate the new domain-slot-value triple and the topic of user utterance that we introduce into the belief state, respectively. Words in yellow illustrate the topics of documents that we extract through preprocessing.

To address our defined task, we propose a task-oriented dialog system with **S**emi-Structured **Know**ledge management (SeKnow). It extends the belief state to handle TODs grounded on semi-structured knowledge, and further uses the extended

This paper is a further development of our prior work [1] accepted to ACL-IJCNLP 2021 Findings.
†Corresponding author.

belief state to perform both structured query and document retrieval, whose outputs are thereby used to generate the final response. As a further development of our prior work HyKnow [1], SeKnow fuses the management of structured and unstructured knowledge via their shared domains and entities. Through the knowledge management fusion, the query of structured data can facilitate the retrieval of documents.

To investigate the trade-off between model performance and computational cost, we introduce two implementations of SeKnow based on a non-pretrained **S**equence-**to**-**S**equence [10] model (SeKnow-S2S) and a **P**retrained **L**anguage **M**odel [11], [12] (SeKnow-PLM), respectively. Both implementations are in end-to-end frameworks, where dialog modeling grounded on structured and unstructured knowledge can be jointly optimized to get overall better performance. In SeKnow-S2S, following our prior work HyKnow [1], we apply two schemes of extended belief state decoding to investigate the correlation of structured and unstructured knowledge management. However, different from HyKnow which uses GRU [13] as backbone, we implement SeKnow-S2S based on Universal Transformer [14] instead, which yields better TOD modeling performance. In SeKnow-PLM, we facilitate our system with large-scale pretraining models to further improve its performance, which is a brand new extension of HyKnow.

We evaluate our system on the modified version of MultiWOZ 2.1 [15] dataset, where dialogs involve knowledge in both structured and unstructured forms. Experimental results show that SeKnow-PLM and SeKnow-S2S outperform existing TOD systems with and without model pretraining, respectively, no matter whether those TOD systems add extra components of unstructured knowledge management or not. SeKnow also has strong belief tracking and document retrieval performances, compared to the pipeline knowledge management schemes.

Our contributions are summarized as below:

- We formulate a task of modeling TOD grounded on knowledge with both structured and unstructured contents, to incorporate more domain knowledge into the TOD systems.
- We propose a TOD system SeKnow to address our proposed task, with two implementations SeKnow-S2S and SeKnow-PLM. Both use an extended belief state to manage semi-structured knowledge, and the end-to-end manner to jointly optimize dialog modeling grounded on structured and unstructured knowledge.
- Experimental results show that SeKnow has strong performance in TOD modeling grounded on semi-structured knowledge.

II. Related Work

TOD systems usually use belief tracking, i.e. dialog state tracking (DST), to trace the user goals as *belief states* through multiple dialog turns [16], [17]. The states are converted into a representation of constraints based on different schemes to query the databases [18]–[21]. The entity matching results are then used to generate the system response.

To reduce deployment cost and error propagation, end-to-end trainable networks [22] are introduced into TOD systems, which have continually been studied recently. Typical end-to-end TOD systems include those with structured fusion networks [3], [23], and those with multi-stage sequence-to-sequence framework [2], [4], [24], [25]. With the boom of Transformers [26] and its large-scale pretraining [11], [12], TOD systems based on auto-regressive language modeling have also been developed [5]–[7], which achieve strong TOD modeling performance.

With the development of intelligent assistants, the system should have a good command of massive external knowledge to better accomplish complicated user goals and improve user satisfaction. To realize this, some researchers [27]–[29] equip the system with chatting capability to address both task and non-task content in TODs. Other studies apply knowledge graph [30], [31] or tables via SQL [32] to enrich the knowledge of TOD systems. However, all these studies are still limited in dialog modeling grounded on structured knowledge.

There are a few studies to integrate unstructured knowledge into TOD modeling recently. Doc2Dial [9] formulates document-grounded dialog for information seeking tasks. Beyond Domain APIs [15] introduces knowledge snippets to answer follow-up questions out of the coverage of databases, which provides benchmarks for the 9th Dialog System Technology Challenge (DSTC9) Track-1 task [33] and prompts some further work [34]–[37]. However, they only focus on dialog modeling grounded on unstructured knowledge instead. In this paper, we aim to fill the gap of managing domain-specific knowledge with various sources and structures in end-to-end TOD systems.

III. Task Definition

In this section, we introduce our formulation of modeling TOD grounded on semi-structured knowledge. In particular, we formulate a turn-level TOD modeling task with access to a semi-structured knowledge base, which contains lists of entities characterized by different domains. Each entity has structured attributes (e.g. name and address), and may also be associated with unstructured documents, as shown in Fig. 1. Our TOD modeling task involves both structured and unstructured contents of the knowledge base.

In each dialog turn, the system needs to track the user goals associated with structured knowledge as domain-slot-value triples in the belief state, and then query the structured contents of knowledge base to guide the generation of response. In particular, we denote the user utterance and the system response at turn t as U_t and R_t, respectively. Given the dialog context $C_t = [U_{t-k}, R_{t-k}, ..., U_t]$ where k is the context window size, the system needs to generate current belief state B_t directly or by updating previously generated belief state B_{t-1}, which are formulated as $B_t = f_b(C_t)$ or $B_t = f_b(C_t, B_{t-1})$, respectively. Then the system uses B_t to query the structured attributes of each entity in the knowledge base, and get the query result M_t showing matched entities, formulated as $M_t = f_m(B_t)$.

Besides the structured query, the system also needs to retrieve the unstructured contents of knowledge base according to the user goals to find relevant references for generating the

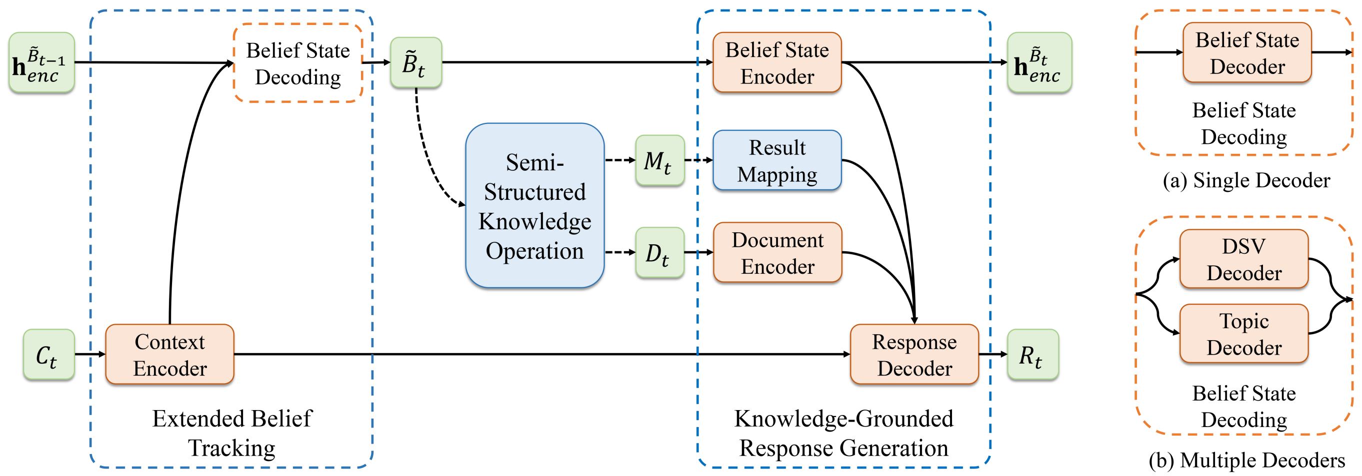

Fig. 2. Overview of SeKnow-S2S. Solid arrows denote the input/output of the encoders or decoders. Dashed arrows denote the knowledge operation or the mapping of query result. C_t, M_t, D_t and R_t represent turn t's dialog context, structured query result, retrieved relevant document and system response, respectively. $\widetilde{B}_t$ and $\mathbf{h}_{enc}^{\widetilde{B}_t}$ denote the extended belief state and its hidden states at turn t. The decoding of $\widetilde{B}_t$ (orange dashed box) is implemented in two different ways: (a) using a single decoder to generate the whole state, and (b) using two decoders to generate the domain-slot-value (DSV) triples and the topic of user utterance separately.

response. Specifically, given the dialog context C_t, the system needs to retrieve the unstructured documents of each entity in the knowledge base and select a relevant document D_t if there exists one. The generated belief state B_t can be optionally used to facilitate the document retrieval, formulated as $D_t = f_d(C_t)$ or $D_t = f_d(C_t, B_t)$.

Finally, the system needs to generate the response R_t based on the dialog context C_t, the belief state B_t, the structured query result M_t and the retrieved document D_t, which is formulated as $R_t = f_r(C_t, B_t, M_t, D_t)$.

IV. Proposed Framework

We propose a task-oriented dialog system with semi-structured knowledge management, **SeKnow**, which addresses our defined task in three steps. First, it uses **extended belief tracking** to track user goals through dialog turns that involve semi-structured knowledge. Secondly, it performs **semi-structured knowledge operation** based on the extended belief state, to search both structured and unstructured knowledge that is relevant to the user goals. Finally, it uses the extended belief state and relevant knowledge to perform the **knowledge-grounded response generation**. Fig. 2 and 3 show our two implementations of SeKnow based on non-pretrained sequence-to-sequence (Seq2Seq) model [10] and pretrained language model (LM) [11], [12], denoted as **SeKnow-S2S** and **SeKnow-PLM**, respectively. SeKnow-S2S is a light-weight Seq2Seq model which requires less computational resources than SeKnow-PLM, while SeKnow-PLM uses large-scale pretraining to yield better TOD modeling performance.

A. Extended Belief Tracking

1) Belief State Extension: We define an extended belief state $\widetilde{B}_t$ which is applicable to track user goals in TODs that involve semi-structured knowledge. Specifically, to capture user goals associated with structured knowledge, $\widetilde{B}_t$ contains the domain-slot-value triples of original B_t. While in dialog turns where user goals involve unstructured knowledge, $\widetilde{B}_t$ has an additional slot *ruk* to indicate that current turn **r**equires **u**nstructured **k**nowledge. The prefix and value of the slot *ruk* represent the involved domain and entity, e.g. *restaurant-ruk*: *Pizza Hut* colored in red in Fig. 1. We denote the combination of original and newly introduced domain-slot-value triples as DSV_t. Following the work of Sequicity [24], we format DSV_t as a text span $[DSV_{t,0}, DSV_{t,1}, ..., DSV_{t,l_t^S-1}]$ with length l_t^S to make it suitable as the input/output of our system, e.g. *restaurant* { *food* = *italian* , *area* = *center* }. Besides, the *topic* of U_t is abstracted in $\widetilde{B}_t$ as a word sequence $T_t = [T_{t,0}, T_{t,1}, ..., T_{t,l_t^T-1}]$ with length l_t^T in each turn related to unstructured knowledge, e.g. *favorite* colored in blue in Fig. 1. $\widetilde{B}_t = [\widetilde{B}_{t,0}, \widetilde{B}_{t,1}, ..., \widetilde{B}_{t,l_t^B-1}]$ is finally the concatenation of DSV_t and T_t, i.e. $[DSV_{t,0}, ..., DSV_{t,l_t^S-1}, T_{t,0}, ..., T_{t,l_t^T-1}]$, where $l_t^B = l_t^S + l_t^T$. In this paper, we define that $X_{t,0:y-1}$ denotes all tokens of X_t before position y, i.e. $[X_{t,0}, X_{t,1}, ..., X_{t,y-1}]$.

2) Extended Belief State Decoding: Our two system implementations SeKnow-S2S and SeKnow-PLM decode $\widetilde{B}_t$ in different ways, which are described as below.

In light-weight SeKnow-S2S, to reduce the complexity of TOD modeling, we decode $\widetilde{B}_t$ on the basis of previous turn's generated state $\widetilde{B}_{t-1}$. We choose a small dialog context window where C_t includes only previous system response R_{t-1} and current user utterance U_t, because $\widetilde{B}_{t-1}$ already summarizes the information in utterances before R_{t-1}. Specifically, following Seq2Seq framework, we first use the *context encoder* to encode C_t, and then decode $\widetilde{B}_t$ based on the hidden states of context encoder $\mathbf{h}_{enc}^{C_t}$ and previous extended belief state $\mathbf{h}_{enc}^{\widetilde{B}_{t-1}}$. Noticing that DSV_t and T_t are grounded on quite different vocabularies, we consider decoding $\widetilde{B}_t$ under two schemes: (a) using the belief state decoder to generate the whole $\widetilde{B}_t$, and (b) using the DSV decoder and the topic decoder to generate

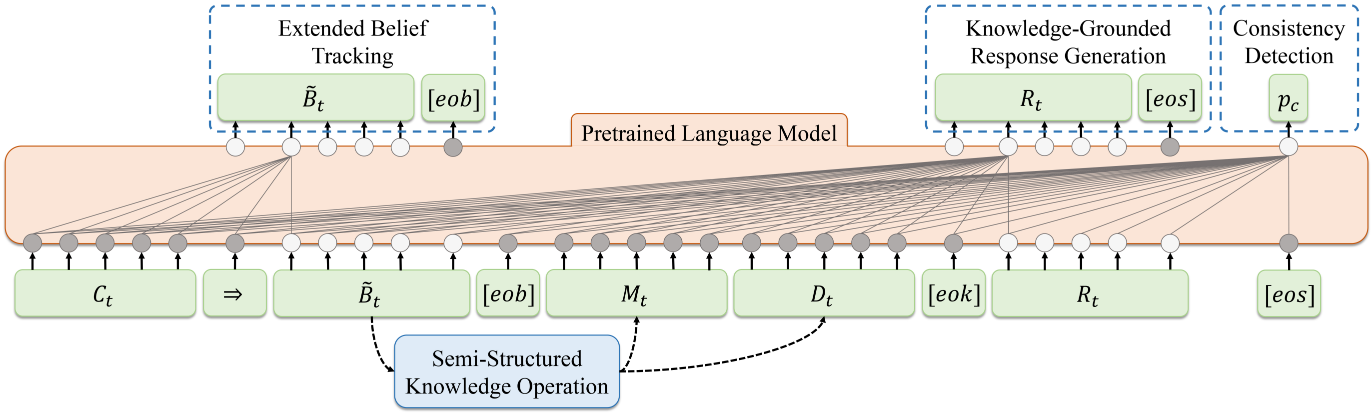

Fig. 3. Overview of SeKnow-PLM. Solid arrows denote the input/output of the pretrained language model. Dashed arrows denote the knowledge operation. Gray lines show the variable dependency of each language modeling sub-task. C_t, $\widetilde{B}_t$, M_t, D_t and R_t represent turn t's dialog context, extended belief state, structured query result, retrieved relevant document and system response, respectively. $=>$, $[eob]$ are shift tokens indicating the start and end of belief tracking, while $[eok]$, $[eos]$ are shift tokens indicating the start and end of response generation. p_c represents the final output probability of dialog consistency.

DSV_t and T_t separately. Each implementation has its own advantages over the other.

Under the single-decoder scheme, the decoding of DSV_t and T_t can be jointly optimized via shared parameters:

$$\begin{aligned} \mathbf{h}_{enc}^{C_t} &= \text{Encoder}^{(\text{C})}(C_t), \\ \widetilde{B}_{t,i} &= \text{Decoder}^{(\text{B})}(\widetilde{B}_{t,0:i-1} | \mathbf{h}_{enc}^{C_t}, \mathbf{h}_{enc}^{\widetilde{B}_{t-1}}), \end{aligned} \tag{1}$$

where $i = 0, 1, ..., l_t^B - 1$.

While under the multi-decoder scheme, the decoding of DSV_t and T_t are fitted to their own smaller decoding spaces (vocabularies), and thus the generation of $\widetilde{B}_t$ can be decomposed into two simpler decoding processes:

$$\begin{aligned} \mathbf{h}_{enc}^{C_t} &= \text{Encoder}^{(\text{C})}(C_t), \\ DSV_{t,j} &= \text{Decoder}^{(\text{S})}(DSV_{t,0:j-1} | \mathbf{h}_{enc}^{C_t}, \mathbf{h}_{enc}^{\widetilde{B}_{t-1}}), \\ T_{t,k} &= \text{Decoder}^{(\text{T})}(T_{t,0:k-1} | \mathbf{h}_{enc}^{C_t}, \mathbf{h}_{enc}^{\widetilde{B}_{t-1}}), \\ \widetilde{B}_t &= [DSV_t, T_t], \end{aligned} \tag{2}$$

where $j = 0, 1, ..., l_t^S - 1$ and $k = 0, 1, ..., l_t^T - 1$.

In SeKnow-PLM where large-scale pretraining is used to get strong TOD modeling capability, we decode $\widetilde{B}_t$ directly based on C_t to avoid the error propagation from previous state $\widetilde{B}_{t-1}$. Since $\widetilde{B}_{t-1}$ is not used in the decoding of $\widetilde{B}_t$, we choose a large context window where C_t includes the whole dialog history, i.e. $[U_0, R_0, ..., U_t]$. We insert shift tokens $=>$ and $[eob]$ before and after $\widetilde{B}_t$ to indicate the start and end of belief state decoding, respectively. Based on C_t, we then use the pretrained language model (PLM) to generate $\widetilde{B}_t$ in the left-to-right auto-regressive manner:

$$\widetilde{B}_{t,i} = \text{PLM}(C_t, \widetilde{B}_{t,0:i-1}), \tag{3}$$

where $i = 0, 1, ..., l_t^B - 1$.

B. Semi-Structured Knowledge Operation

Based on the extended belief state $\widetilde{B}_t$, we conduct both structured data query and unstructured document retrieval in the knowledge base, whose outputs M_t and D_t are used to guide the generation of system response. The structured query can facilitate the unstructured document retrieval by helping to identify the relevant entity and narrow down the document candidates. This is different from our prior work HyKnow [1] where structured and unstructured knowledge are operated independently of each other.

We first query the knowledge base to select entities whose domains and structured attributes **exact match** the triples DSV_t in $\widetilde{B}_t$. In dialog turns that involve unstructured knowledge, the value of slot *ruk* in DSV_t serves to **fuzzy match** the name or ID of each entity, and only the best-matched entity is selected. Similar to DSV_t, we format the query result M_t as a text span indicating the number of matched entities in each domain, e.g. *restaurant 2 match* , *train no match*. We only need the number of matched entities instead of their specific information, because we consider generating delexicalized responses with specific slot values replaced by placeholders to improve data efficiency [38].

Based on the structured query result, we then perform document retrieval to find unstructured knowledge relevant to C_t. Specifically, we first preprocess the documents of each entity in the knowledge base, to extract each document's topic as its retrieval index[1], e.g. *vegetarian* and *favorite* colored in yellow in Fig. 1. Then we retrieve the documents of the best-matched entity selected in the structured query to find the relevant document D_t, where we use the topic T_t in $\widetilde{B}_t$ to **fuzzy match** the topic of each document and choose the best-matched one as D_t. Noting that D_t is set to *none* if the triple with slot *ruk* or the topic of user utterance is not available, i.e. unstructured knowledge is not required at turn t.

C. Knowledge-Grounded Response Generation

We generate the system response R_t, i.e. a word sequence $[R_{t,0}, R_{t,1}, ..., R_{t,l_t^R-1}]$ with length l_t^R, based on the dialog context C_t, the extended belief state $\widetilde{B}_t$, and the outputs of semi-structured knowledge operation M_t and D_t. Our two

[1]See Appendix A for more details of the document preprocessing.

system implementations SeKnow-S2S and SeKnow-PLM also decode R_t in different ways, which are described as below.

In SeKnow-S2S, we first use the same context encoder in Sec. IV-A2 to encode C_t into hidden states $\mathbf{h}_{enc}^{C_t}$. Besides, we use the *belief state encoder* and the *document encoder* to encode $\widetilde{B}_t$ and D_t into hidden states $\mathbf{h}_{enc}^{\widetilde{B}_t}$ and $\mathbf{h}_{enc}^{D_t}$, respectively. For the structured query result M_t, we follow MultiWOZ [19] to map it to a vector $\mathbf{m}_t$ according to the number of matched entities in turn t's active domain and whether the booking is available. Based on the hidden states of all the encoders and the vector $\mathbf{m}_t$, we use the *response decoder* to generate R_t, formulated as:

$$\begin{aligned}
\mathbf{h}_{enc}^{\widetilde{B}_t} &= \mathrm{Encoder}^{(\mathrm{B})}(\widetilde{B}_t),\\
\mathbf{h}_{enc}^{D_t} &= \mathrm{Encoder}^{(\mathrm{D})}(D_t),\\
\mathbf{m}_t &= \mathrm{Mapping}(M_t),\\
R_{t,i} &= \mathrm{Decoder}^{(\mathrm{R})}(R_{t,0:i-1}|\mathbf{h}_{enc}^{C_t}, \mathbf{h}_{enc}^{\widetilde{B}_t}, \mathbf{h}_{enc}^{D_t}, \mathbf{m}_t),
\end{aligned} \tag{4}$$

where $i = 0, 1, ..., l_t^R - 1$.

In SeKnow-PLM, we insert shift tokens $[eok]$ and $[eos]$ before and after R_t to indicate the start and end of response decoding, respectively. Based on the concatenation of C_t, $\widetilde{B}_t$, M_t and D_t, we use the pretrained language model (PLM) to generate R_t in the left-to-right auto-regressive manner:

$$R_{t,i} = \mathrm{PLM}(C_t, \widetilde{B}_t, M_t, D_t, R_{t,0:i-1}), \tag{5}$$

where $i = 0, 1, ..., l_t^R - 1$.

D. Consistency Detection

In SeKnow-PLM, we also consider an auxiliary consistency detection task [6], [11] for model training. Specifically, we follow AuGPT [7] to randomly corrupt half of the dialog samples and train our model to detect each sample's consistency. The corruption includes three types happening with equal probability: (a) $\widetilde{B}_t$ is wholly replaced by another, (b) The value of each slot in $\widetilde{B}_t$ is replaced by a different one. (c) R_t is replaced by another. We apply a binary classifier on the output hidden state of final token $[eos]$ to predict the probability of dialog consistency p_c, formulated as:

$$p_c = \mathrm{PLM}(C_t, \widetilde{B}_t, M_t, D_t, R_t). \tag{6}$$

E. Model Training and Implementation Details

In SeKnow-S2S, we use Universal Transformers (UT) [14] to implement our encoders and decoders. Unlike the original Transformers [26] which stacks multiple layers, UT builds the encoder/decoder as a single self-attention & feed-forward layer with recurrent connection, combining a two-dimensional (position, time) coordinate embeddings. Besides, UT improves the feed-forward network with depth-wise separable convolutions [39] instead of linear transformations. As a generalization of Transformers, UT remedies the deficiencies of Transformers in some Seq2Seq tasks, especially the copy [40] of text which plays a significant role in TOD modeling [24].

In SeKnow-PLM, we use pretrained GPT-2 [12] as our model backbone. Besides, we follow AuGPT [7] to further pretrain GPT-2 on large-scale TOD corpus Taskmaster-1 [41] and Schema-Guided Dialogue [20].

Both SeKnow-S2S and SeKnow-PLM are optimized via supervised training. In particular, each dialog turn in the training data is initially labeled with the original belief state and the relevant document. We extend the belief state label based on the domain, entity and extracted topic of the relevant document. Then the extended belief state label and the reference response are used to calculate the cross-entropy loss with the generated $\widetilde{B}_t$ and R_t, respectively, formulated as:

$$\mathcal{L}_B = -\mathrm{log}p(\widetilde{B}_t|C_t) = -\sum_{i=0}^{l_t^B} \mathrm{log}p(\widetilde{B}_{t,i}|C_t, \widetilde{B}_{t,0:i-1}). \tag{7}$$

$$\begin{aligned}
\mathcal{L}_R &= -\mathrm{log}p(R_t|C_t, \widetilde{B}_t, M_t, D_t)\\
&= -\sum_{i=0}^{l_t^R} \mathrm{log}p(R_{t,i}|C_t, \widetilde{B}_t, M_t, D_t, R_{t,0:i-1}).
\end{aligned} \tag{8}$$

In SeKnow-PLM whose training contains auxiliary consistency detection task, we also calculate the binary cross-entropy loss on the output probability of dialog consistency p_c:

$$\mathcal{L}_C = -y_c \mathrm{log}p_c - (1 - y_c)\mathrm{log}(1 - p_c). \tag{9}$$

where y_c is the label indicating whether the contents of current dialog sample are consistent ($y_c = 1$) or not ($y_c = 0$).

We sum all losses together and perform gradient descent in each turn to optimize the model parameters.[2]

V. Experimental Settings

A. Dataset

We evaluate our proposed system on the modified MultiWOZ 2.1 [15] dataset, where crowd-sourcing workers are hired to insert additional turns into the original MultiWOZ dialogs. Each newly inserted turn involves unstructured knowledge in one of the four domains (restaurant, hotel, taxi, train), and is annotated with its relevant document. All documents are characterized by different domains and entities in an additional document base. We fuse the document base and the original MultiWOZ database via their shared domains and entities, to get our semi-structured knowledge base. Noting that the other three MultiWOZ domains (attraction, hospital, police) are not involved in the unstructured documents.[3]

B. Baselines

We compare SeKnow with **1) existing end-to-end (E2E) TOD models and dialog state tracking (DST) models**, to explore the benefits of incorporating unstructured knowledge management into TOD modeling. We also compare SeKnow with **2) unstructured knowledge management models**, to investigate our system's document retrieval performance. For the comparison with pipeline systems which have both structured and unstructured knowledge management, we also consider **3) the combinations of 1) and 2)** as our baselines. Besides, we compare SeKnow with **4) our prior work HyKnow**, to show the effectiveness of our system improvements.

[2]See Appendix B for more model training and implementation details.

[3]See Appendix C for details of data statistics.

1) E2E TOD Models and DST Models: We consider two light-weight baseline E2E TOD models with different types of structures: **UniConv** [3] uses a structured fusion [23] design, while **LABES-S2S** [4] is based on a multi-stage Seq2Seq [24] architecture. Besides, we consider two large-scale baseline E2E TOD models developed from pretrained GPT-2 [12]: **SimpleTOD** [5] directly finetunes GPT-2 to model TOD in the auto-regressive manner, while **AuGPT** [7] further pretrains GPT-2 on large TOD corpus before finetuning and applies training data augmentation based on back-translation [42]–[44]. All four E2E TOD models only manage structured knowledge (database) in their TOD modeling. In addition to E2E TOD models, we also compare SeKnow with existing DST models in the belief tracking evaluation. Specifically, we use **TRADE** [45] and **TripPy** [46] as two DST baselines, which are representative BERT-free and BERT-based [11] DST models, respectively.

2) Unstructured Knowledge Management Models: We first compare SeKnow with **Beyond Domain APIs (BDA)** [15], which uses two classification modules based on BERT [11] to detect dialog turns requiring unstructured knowledge and retrieve relevant documents, respectively. We also compare our system with three representative models developed from BDA in the first track of the 9th Dialog System Technology Challenge (DSTC9). Thulke et al. [36] propose two models based on RoBERTa [47] to better address the document retrieval problem in BDA: **Hierarchical Knowledge Selection (HKS)** model narrows down the candidates of document retrieval by first identifying the relevant domain and entity, while **Dense Knowledge Retrieval (DKR)** model improves the efficiency of document retrieval by formulating it as a metric learning problem. Kim et al. [37] propose an **End-to-End Document-Grounded Conversation (E2E-DGC)** model based on T5 [48] to optimize the document retrieval jointly with the knowledge-grounded response generation, which also yields better performance than BDA. Moreover, we use standard information retrieval (IR) systems **TF-IDF** [49] and **BM25** [50] as the other two baseline models.

3) Combinations: We combine the unstructured knowledge management model BDA or HKS with every DST or E2E TOD model. Specifically, BDA or HKS detects dialog turns involving unstructured knowledge, and generates responses in these turns based on the dialog context and retrieved documents. While the DST or E2E TOD model handles the rest dialog turns which are only related to structured knowledge. Noting that BDA uses finetuned GPT-2 [12] to generate responses, while HKS finetunes BART [51] instead and follow REALM [52] to apply retrieval augmented response generation.

4) HyKnow: Similar to SeKnow-S2S, HyKnow also extends the belief state to manage both structured and unstructured knowledge, and jointly optimizes TOD modeling grounded on these two kinds of knowledge in an end-to-end Seq2Seq framework. However, HyKnow operates the structured and unstructured knowledge independently of each other, and it implements the Seq2Seq modeling based on traditional GRU [13]. In our experiments, we test HyKnow's performance under the single-decoder belief state decoding scheme, denoted as HyKnow (Single) in our prior work [1].

Noting that DST and E2E TOD models based on BERT or GPT-2 utilizes large-scale language model (LM) pretraining to improve their TOD modeling performance, which however requires large model sizes and computing resources. For fair comparisons, we distinguish them from other non-pretrained light-weight models in our experiments.

VI. Results and Analysis

We compare SeKnow-S2S with light-weight non-pretrained baseline models, and test its performance under both the single-decoder and multi-decoder belief state decoding schemes, denoted as SeKnow-S2S (Single) and SeKnow-S2S (Multiple), respectively. Besides, we compare SeKnow-PLM with baseline models which utilize large-scale pretrained language models (LM). All implementations of SeKnow come to the same conclusions when compared with their corresponding baseline systems, which are described in detail below.

A. End-to-End Evaluation

Table I shows our experimental results of the end-to-end (E2E) TOD evaluation, where we evaluate the task completion rate and language quality of system responses. In terms of the task completion rate, we measure whether the system provides correct entities (**Inform** rate) and answers all the requested information (**Success** rate) in a dialog, following MultiWOZ [19]. For the evaluation of language quality, we adopt commonly used metrics **BLEU** [53], **METEOR** [54] and **ROUGE-L** [55]. Moreover, we use **Combined** score computed by $(Inform + Success) \times 0.5 + BLEU$ for overall evaluation, as suggested by MultiWOZ 2.1 [56].

We find that SeKnow-S2S has significantly better task completion rate compared to the non-pretrained E2E TOD models, which is even better than SimpleTOD based on large-scale pretrained GPT-2. It also generates responses with better language quality than the E2E TOD models. The same conclusions are also observed by comparing SeKnow-PLM with GPT-2 based E2E TOD models. All above results show that our belief state extension helps to distinguish whether a dialog turn does or does not involve unstructured knowledge, which avoids the confusion between handling the two kinds of dialog turns. In addition, we retrieve unstructured documents to provide relevant references for generating the response, which guide our system to give more appropriate responses in dialog turns that involve unstructured knowledge.

We also observe that SeKnow-S2S and SeKnow-PLM outperform the combinations of BDA/HKS with non-pretrained and pretrained E2E TOD models, respectively. This indicates that our end-to-end model framework has advantages over the pipeline structures of combination models. In particular, dialog modeling grounded on the structured and unstructured knowledge are integrated in a uniform architecture in our system, where they are jointly optimized to an overall better performance. Since our system is trained end-to-end, it also has lower deployment cost in real-world applications compared to the pipeline systems.

TABLE I
END-TO-END EVALUATION RESULTS ON MODIFIED MULTIWOZ 2.1. "+" DENOTES THE MODEL COMBINATION. BEST RESULTS AMONG SYSTEMS WITH AND WITHOUT PRETRAINED LANGUAGE MODELS (LM) (I.E. BELOW AND ABOVE THE INTERNAL DIVIDING LINE) ARE SEPARATELY MARKED IN BOLD.

Model	Pretrained LM	Inform	Success	BLEU	METEOR	ROUGE-L	Combined
UniConv	none	71.5	61.8	18.5	37.8	40.5	85.7
UniConv + BDA	-	72.0	62.6	16.9	35.7	38.9	84.2
UniConv + HKS	-	72.8	63.5	17.9	37.2	39.7	86.1
LABES-S2S	none	76.5	65.3	17.8	36.8	39.9	88.7
LABES-S2S + BDA	-	77.1	66.2	15.7	33.8	37.8	87.4
LABES-S2S + HKS	-	78.2	66.7	18.2	37.5	40.1	90.7
SeKnow-S2S (Single)	none	**82.9**	**68.7**	**19.0**	**38.6**	**40.8**	**94.8**
SeKnow-S2S (Multiple)	none	80.6	68.4	18.7	38.1	40.3	93.2
SimpleTOD	GPT-2	81.7	67.9	14.5	34.2	37.0	89.3
SimpleTOD + BDA	-	83.3	68.6	14.8	33.6	36.5	90.8
SimpleTOD + HKS	-	83.5	68.8	16.9	36.4	38.7	93.1
AuGPT	GPT-2	88.9	69.5	16.8	36.1	39.3	96.0
AuGPT + BDA	-	91.2	70.4	16.8	36.0	39.1	97.6
AuGPT + HKS	-	91.6	70.7	17.0	36.3	39.3	98.2
SeKnow-PLM	GPT-2	**93.6**	**71.9**	**17.3**	**36.8**	**40.0**	**100.1**

TABLE II
CONTEXT-TO-RESPONSE GENERATION ON MODIFIED MULTIWOZ 2.1. ALL SYMBOLS AND MARKINGS HAVE SAME MEANING AS IN TABLE I.

Model	Inform	Success	BLEU	Combined
UniConv	84.2	71.8	19.0	97.3
UniConv + HKS	85.8	73.9	**19.5**	99.6
LABES-S2S	83.6	74.2	18.3	97.2
LABES-S2S + HKS	85.0	75.3	19.2	99.4
SeKnow-S2S	**87.6**	**76.8**	**19.5**	**101.7**
SimpleTOD	87.5	76.4	16.3	98.3
SimpleTOD + HKS	89.0	77.2	17.1	100.2
AuGPT	94.2	76.4	17.4	102.7
AuGPT + HKS	95.3	77.6	17.5	104.0
SeKnow-PLM	**96.0**	**78.0**	**17.9**	**104.9**

TABLE III
ORIGINAL TURNS' BELIEF TRACKING RESULTS ON MODIFIED MULTIWOZ 2.1. ALL SYMBOLS AND MARKINGS HAVE SAME MEANING AS IN TABLE I.

Model	Pretrained LM	Joint Goal
TRADE	none	42.9
TRADE + BDA	-	43.8
TRADE + HKS	-	43.9
UniConv	none	45.5
UniConv + BDA	-	46.5
UniConv + HKS	-	47.0
LABES-S2S	none	46.0
LABES-S2S + BDA	-	46.8
LABES-S2S + HKS	-	47.6
SeKnow-S2S (Single)	none	**49.1**
SeKnow-S2S (Multiple)	none	48.4
TripPy	BERT	50.4
TripPy + BDA	-	51.2
TripPy + HKS	-	51.4
SimpleTOD	GPT-2	48.4
SimpleTOD + BDA	-	49.8
SimpleTOD + HKS	-	50.1
AuGPT	GPT-2	55.0
AuGPT + BDA	-	56.0
AuGPT + HKS	-	56.6
SeKnow-PLM	GPT-2	**58.5**

B. Context-to-Response Generation

We also conduct evaluations on the context-to-response (C2R) generation, where systems directly use the oracle belief state and knowledge to generate the response. The experimental results are shown in Table II, where we observe the same conclusions as in the E2E evaluation (Table I). This again shows our system's superiority in TOD modeling grounded on semi-structured knowledge. Noting that we do not differentiate the two belief state decoding schemes of SeKnow-S2S because they share the same substructure in the C2R task. We also do not list the combination models with BDA because their performances are almost the same as those with HKS.

Additionally, we observe that SeKnow's performance gap between E2E and C2R evaluations is smaller than the baseline models, reflected in the smaller variations of the combined score. This shows that the belief state and knowledge provided by our system are probably closer to the oracle and may give stronger guidance to generate the response.

C. Knowledge Management Evaluation

To further investigate our system's E2E performance, we conduct evaluations on the intermediate structured and unstructured knowledge management. In terms of structured knowledge management, we evaluate the belief tracking performance which directly determines the structured data query accuracy. Specifically, we use the **Joint Goal** accuracy [17] to measure whether belief states are predicted correctly in the original dialog turns of MultiWOZ 2.1. Noting that we do not consider the newly inserted dialog turns where belief states are not uniformly defined: SeKnow uses the extended belief state, while baseline DST/E2E models only parse the original belief state, and combination models do not update the belief state. For unstructured knowledge management, we adopt standard information retrieval metrics **R@1** and **MRR@5** to evaluate the document retrieval performance. Table III and IV shows our evaluation results.

TABLE IV
DOCUMENT RETRIEVAL RESULTS ON MODIFIED MULTIWOZ 2.1. BEST RESULTS ARE MARKED IN BOLD.

Model	Type	MRR@5	R@1
TF-IDF	standard IR	68.7	54.1
BM25	standard IR	69.2	52.5
BDA	classification	80.6	69.8
E2E-DGC	classification	89.9	87.1
DKR	dense retrieval	92.9	89.8
HKS	classification	**95.7**	91.9
SeKnow-S2S (Single)	topic match	91.4	89.9
SeKnow-S2S (Multiple)	topic match	90.6	89.0
SeKnow-PLM	topic match	94.7	**93.4**

In terms of belief tracking, SeKnow-S2S and SeKnow-PLM outperform the non-pretrained and pretrained DST/E2E models, respectively. This is because our extended belief tracking can detect the newly inserted turns apart from the original turns (via the slot *ruk*), which improves our system's awareness on deciding when to update the triples related to original dialog process. SeKnow-S2S and SeKnow-PLM also have better belief tracking performance compared to the combinations of BDA/HKS with non-pretrained and pretrained DST/E2E models, respectively. This is because error propagation on updating belief states is eliminated in our system compared to the pipeline framework: The pipeline system either updates the belief state or retrieves the document in one turn, but SeKnow can perform both operations in the nature of its E2E design.

In the document retrieval evaluation, we find that SeKnow has strong performance among the unstructured knowledge management models. In particular, SeKnow-S2S outperforms BDA and standard IR systems, and is comparable with the strong baseline models proposed in the first track of DSTC9, which all utilize large-scale model pretraining. While SeKnow-PLM scores close to the strongest baseline model HKS on MRR@5 metric, and achieves the highest R@1 rate among all the baseline models. Above experimental results indicate that our system's document retrieval scheme with topic matching has advantages over the retrieval schemes of baseline models. Specifically, SeKnow retrieves documents based on the highly simplified semantic information, i.e. the topic, which reduces the complexity of the retrieval process. This makes the retrieval scheme of SeKnow more concise and effective than the baseline models which all directly calculate the relevance of dialog context to every document content.

D. Single vs. Multiple Decoders

In this section, we compare the two extended belief state decoding schemes of our Seq2Seq system implementation SeKnow-S2S. We calculate the vocabularies of DSV triples, the topic and their combination (which are 709, 166 and 862), and observe that the last one approximately equals to the sum of the former two. This confirms our assumption in Sec. IV-A2 that DSV triples used for structured data query and the topic used for unstructured document retrieval have quite different vocabularies, which motivates our proposal of the multi-decoder belief state decoding scheme.

However, we find that SeKnow-S2S (Single) outperforms SeKnow-S2S (Multiple) in both E2E and knowledge management evaluations, as shown in Table I, III and IV. This shows that the decoding of DSV triples and topic can benefit from the joint optimization via shared parameters, although they are grounded on quite different vocabularies. The superiority of joint optimization further implies that the structured and unstructured knowledge management in TOD modeling have a positive correlation, since they commonly involve task-specific domain knowledge and entities. Therefore, the two kinds of knowledge management can learn from each other through joint training, and achieve overall better performance compared to separating them apart. This conclusion also makes it reasonable for our system to fuse the management of structured and unstructured knowledge via their shared domains and entities.

E. SeKnow vs. HyKnow

We compare SeKnow with our prior work HyKnow to explore the benefits of our system improvements. In particular, we follow Sec. VI-A and VI-C to compare the E2E and knowledge management performances of SeKnow-S2S (Single), SeKnow-PLM and HyKnow (Single). We also evaluate their extended part of belief tracking and relevant entity matching performances to further investigate their management of unstructured knowledge. Specifically, we evaluate the above three models' **P**recision, **R**ecall and **F1**-measure of predicting the triple with *ruk* and the topic in our extended belief state. And we follow the document retrieval evaluation to compare their **MRR@5** and **R@1** rates in matching the relevant entity whose unstructured documents are involved. Besides, we compare the sizes and training time of the above three models to investigate their costs of computational resources. The comparison results are shown in Table V and VI.

We first observe that SeKnow-S2S (Single) has almost 10 percent higher MRR@5 and R@1 rates of document retrieval compared to our prior work HyKnow (Single). And we find that SeKnow-S2S (Single) also matches the relevant entity with significantly better MRR@5 and R@1 rates than HyKnow (Single), although their recall of the belief state's extended part (i.e. the triple with *ruk* and the topic) differs much less. The above results indicate that SeKnow has great advantages over HyKnow in finding relevant entities and documents, which largely benefits from the fusion of structured and unstructured knowledge management. Specifically, through the knowledge management fusion, the original triples in SeKnow's belief state can provide more constraints via structured query to help narrow down the candidates of relevant entities and documents. This simplifies the matching of entity name/ID and topic in dialog turns requiring unstructured knowledge, and avoids some document retrieval errors caused by the misprediction of belief state's extended part.

SeKnow-S2S (Single) also outperforms HyKnow (Single) in extended belief tracking, with higher Joint Goal accuracy in predicting belief state's original triples and better F1-measure in predicting belief state's extended part. The superiority of SeKnow-S2S (Single) in extended belief tracking also

TABLE V
ABLATION STUDY RESULTS AND COMPARISONS OF OUR PROPOSED MODELS ON KNOWLEDGE MANAGEMENT AND END-TO-END PERFORMANCES.

Model	MRR@5	R@1	Joint Goal	Inform	Success	BLEU	METEOR	ROUGE-L	Combined
HyKnow (Single)	81.7	80.2	48.0	81.9	68.3	19.0	38.5	40.9	94.1
SeKnow-S2S (Single)	91.4	89.9	49.1	82.9	68.7	19.0	38.6	40.8	94.8
- w/o KM Fusion	84.0 (-7.4)	81.7 (-8.2)	49.1 (-0.0)	82.9	68.7	18.3	37.8	40.2	94.1 (-0.7)
- w/o Joint Optim	82.2 (-9.2)	79.4 (-10.5)	46.9 (-2.2)	79.2	65.6	18.5	38.1	39.6	90.9 (-3.9)
SeKnow-PLM	94.7	93.4	58.5	93.6	71.9	17.3	36.8	40.0	100.1
- w/o KM Fusion	91.0 (-3.7)	88.9 (-4.5)	58.5 (-0.0)	93.6	71.9	16.5	36.1	39.5	99.3 (-0.8)
- w/o Joint Optim	90.2 (-4.5)	86.9 (-6.5)	57.1 (-1.4)	92.4	70.3	16.7	36.0	39.2	98.1 (-2.0)

TABLE VI
COMPARISONS OF OUR PROPOSED MODELS ON EXTENDED PART OF BELIEF TRACKING, RELEVANT ENTITY MATCHING AND COMPUTATIONAL COST.

Model	Triple with *ruk*			Topic			Entity Matching		Model Size	Training Time
	P	R	F1	P	R	F1	MRR@5	R@1		
HyKnow (Single)	98.6	79.6	88.1	99.3	87.7	93.1	86.1	83.8	4.08M	767mins (1 GPU)
SeKnow-S2S (Single)	98.9	81.4	89.3	99.5	88.2	93.5	94.0	91.5	43.7M	368mins (1 GPU)
SeKnow-PLM	99.4	88.0	93.4	99.6	94.1	96.8	96.4	94.4	124M	714mins (4 GPUs)

contributes to its better task completion rate (i.e. Inform and Success rate) than HyKnow (Single) in E2E evaluation. The above results prove that Universal Transformers used to implement SeKnow has stronger Seq2Seq modeling ability in TOD, compared to GRU networks used to implement HyKnow. Besides, SeKnow-S2S (Single) requires less training time than HyKnow (Single) under the same GPU usage, although the former has much larger model size than the latter. This shows that Universal Transformers also has better training efficiency than GRU in terms of TOD modeling.

With the usage of pretrained language model, SeKnow-PLM achieves significantly better performances than HyKnow (Single) in terms of extended belief tracking, document retrieval and task completion. It also obviously outperforms our non-pretrained system implementation SeKnow-S2S (Single) in all above three aspects. In trade-off, SeKnow-PLM has a larger model size and requires more GPUs to get comparable training time with HyKnow (Single). This shows that large-scale model pretraining has great power in improving TOD modeling grounded on semi-structured knowledge, while at the cost of extra pretraining corpus and more computational resources. Noting that although SeKnow-PLM does not score higher than HyKnow (Single) and SeKnow-S2S (Single) on BLEU, METEOR and ROUGE-L, the three metrics can not absolutely represent the language quality of system response. Through the human evaluation in Sec. VI-H, we prove that SeKnow-PLM actually has better response quality than our non-pretrained models.

F. Ablation Study

We ablate 1) the fusion of structured and unstructured knowledge management (KM), or further 2) the joint optimization of structured and unstructured knowledge-grounded TOD modeling, to investigate their respective roles in our system, denoted as w/o KM Fusion and w/o Joint Optim in Table V. Both ablations separate apart the structured and unstructured knowledge, i.e. we split the semi-structured knowledge base back into the original database and document base in modified MultiWOZ 2.1. In the ablation of KM Fusion, we use the original and extended parts of belief state to separately perform structured database query and unstructured document retrieval, where the value of slot *ruk* and the topic are used to retrieve the documents of all entities in the active domain. Further in the ablation of Joint Optim, we train two SeKnow models to address our TOD modeling task: one tracks the original belief state, performs database query and generates responses in original dialog turns, while the other parses the extended part of belief state, performs document retrieval and generates responses in newly inserted dialog turns. In testing, we use the absence or presence of slot *ruk* to judge whether a dialog turn belongs to original or newly inserted turns.

Without knowledge management fusion (w/o KM Fusion), SeKnow suffers from evident performance declines in terms of document retrieval, which leads to lower language quality of generated responses. This again shows that the fusion of knowledge management has great benefits to system's document retrieval performance, where the structured data query can facilitate the matching of relevant entities and documents.

We also observe that removing joint optimization (w/o Joint Optim) brings SeKnow evident performance declines in the E2E evaluation. This suggests that joint optimization plays a significant role in improving SeKnow's E2E performance, where TOD modeling grounded on structured and unstructured knowledge can benefit each other by learning shared parameters. The ablation of joint optimization also causes further declines in SeKnow's knowledge management performance, compared to that without KM Fusion. This again indicates that structured and unstructured knowledge management are positively correlative and can get benefit from joint training.

G. Between Structured and Unstructured Knowledge

In this section, we investigate how the newly inserted dialog turns (involving unstructured knowledge) affect systems' TOD performances (i.e. tracking user goals associated with

TABLE VII
BELIEF TRACKING AND END-TO-END EVALUATION RESULTS ON THE ORIGINAL AND MODIFIED MULTIWOZ 2.1 TEST SET. THE EVALUATION IS CONDUCTED ONLY IN THE ORIGINAL DIALOG TURNS.

Test Set	Model	Joint Goal	Inform	Success	BLEU	METEOR	ROUGE-L	Combined
Original	LABES-S2S + HKS	49.3	82.3	69.7	17.8	37.1	40.2	93.8
	AuGPT + HKS	58.5	93.4	73.5	17.2	36.5	40.0	100.7
	SeKnow-S2S (Single)	49.9	83.5	69.3	18.9	38.2	41.0	95.3
	SeKnow-PLM	58.9	95.6	72.4	17.9	36.8	40.2	101.9
Modified	LABES-S2S + HKS	47.6 (-1.7)	78.2	66.7	17.6	36.8	39.6	90.1 (-3.7)
	AuGPT + HKS	56.6 (-1.9)	91.6	70.7	16.9	36.3	39.3	98.1 (-2.6)
	SeKnow-S2S (Single)	49.1 (-0.8)	82.9	68.7	18.2	37.3	40.5	94.0 (-1.3)
	SeKnow-PLM	58.5 (-0.4)	93.6	71.9	17.5	36.6	40.0	100.3 (-1.6)

TABLE VIII
HUMAN EVALUATION RESULTS ON MODIFIED MULTIWOZ 2.1, RESULTS IN ORIGINAL AND NEWLY INSERTED TURNS ARE SHOWN SEPARATELY.

Model	Original			Newly Inserted		
	Cohe.	Info.	Corr.	Cohe.	Info.	Corr.
HyKnow (Single)	2.58	2.52	2.44	2.56	2.34	2.46
SeKnow-S2S (Single)	2.60	2.56	2.48	2.58	2.50	2.54
AuGPT	2.66	2.68	2.60	2.62	2.54	2.58
AuGPT + HKS	**2.68**	2.66	2.64	2.64	2.66	2.62
SeKnow-PLM	2.64	**2.70**	**2.68**	**2.66**	**2.70**	**2.72**

structured knowledge and generating responses) in the original dialog turns. Specifically, we evaluate systems' belief tracking and E2E performances on both the original and modified MultiWOZ 2.1 test sets. This evaluation is conducted only in the original dialog turns, which is different from the E2E evaluation conducted in all turns (Table I). We evaluate our non-pretrained and pretrained system implementations SeKnow-S2S (Single) and SeKnow-PLM, compared with their corresponding strong baseline models LABES-S2S + HKS and AuGPT + HKS. The results of this experiment are shown in Table VII, where both comparisons come to the same conclusions described as below.

We first find that all the models' performances are degraded when transferred from the original to the modified MultiWOZ 2.1 test set. This indicates that the newly inserted turns involving new domain knowledge may interrupt the original dialogs, which complicates the dialog process and causes difficulties in the original turns' dialog modeling.

However, we observe that SeKnow suffers from less reduction compared to the baseline combination models. This shows that our system has a stronger resistance to the interruptions of newly inserted dialog turns, which benefits from our end-to-end modeling. Specifically, SeKnow jointly optimizes dialog modeling of the original and newly inserted dialog turns in a uniform end-to-end framework. This unified modeling approach improves our system's flexibility in switching between the two kinds of turns, and thus makes it more competent in handling the complicated dialog process.

H. Human Evaluation

There is still a gap between the evaluation results of automatic metrics and the real E2E performance of TOD systems. Therefore, we conduct human evaluation to more adequately test our system's E2E performance. In particular, we test the performance of SeKnow-PLM which achieves the best combined score in the automatic E2E evaluation, compared with the strongest E2E baseline AuGPT and its combination with HKS. We also compare the above models with our non-pretrained system implementation SeKnow-S2S (Single) and our prior work HyKnow (Single), to further investigate the effectiveness of model pretraining in E2E TOD modeling.

We conduct human evaluation separately on the two types (original and newly inserted) of dialog turns. Specifically, we sample fifty dialog turns of each type and ask the judges to evaluate each turn's system response on three aspects. **Coherence (Cohe.)** measures how well the response is coherent with the dialog context. **Informativeness (Info.)** measures how well the response can provide sufficient information that meets the user requests. **Correctness (Corr.)** measures how well the information in response is consistent with the ground truth knowledge, i.e. relevant entities' attributes and documents. All the three aspects are scored on a Liker scale of 1-3, which denotes *bad*, *so-so* and *good*. Table VIII shows our human evaluation results.

We find that SeKnow-PLM, AuGPT and AuGPT + HKS significantly outperform SeKnow-S2S (Single) and HyKnow (Single) in both original and newly inserted dialog turns, especially in terms of informativeness and correctness. This indicates that large-scale model pretraining has great benefits in promoting the response quality in E2E TOD modeling, although pretrained LM does not bring higher scores of language quality (i.e. BLEU, METEOR and ROUGE-L) in automatic E2E evaluation (Table I).

Compared to AuGPT and AuGPT + HKS, SeKnow-PLM scores higher on informativeness and correctness in the original dialog turns. This is consistent with the automatic evaluations, showing that SeKnow-PLM has better belief tracking and task completion performances. SeKnow-PLM also outperforms AuGPT and AuGPT + HKS in the newly inserted dialog turns. Specifically, SeKnow-PLM generates responses with significantly better informativeness and correctness than AuGPT. This again shows that the management of unstructured knowledge is beneficial for generating appropriate responses. Compared to AuGPT + HKS, the responses generated by SeKnow-PLM also achieve much better correctness, which benefits from our model's higher document retrieval accuracy (i.e. higher R@1 rate in Table IV).

TABLE IX
COMPARISONS OF BELIEF TRACKING, DOCUMENT RETRIEVAL AND RESPONSE GENERATION RESULTS WHEN DIALOG STEPS INTO A TURN INVOLVING UNSTRUCTURED KNOWLEDGE. "HUMAN" DENOTES GROUND TRUTH RESULTS.

U_1: We are looking for **free wifi** and a nice **4 star guesthouse** that has **free parking**. We do not care about area or price range. B_1: hotel-internet: yes, hotel-parking: yes, hotel-stars: 4, hotel-type: guesthouse R_1: I would recommend the **Acorn Guest House**. It is a moderately priced guesthouse in the north. Would you like me to book a room? U_2: Do they provide an **Italian breakfast**?	
Human	B_2: hotel-internet: yes, hotel-parking: yes, hotel-stars: 4, hotel-type: guesthouse D_2: Continental, vegetarian, and a full English/Irish **breakfast** are available at **Acorn Guest House**. R_2: **No**, they do not but they have Irish breakfast. Would you like me to go ahead and book it?
AuGPT	B_2: **restaurant-food: Italian**, hotel-internet: yes, hotel-parking: yes, hotel-stars: 4, hotel-type: guesthouse R_2: **Da Vinci Pizzeria** is an Italian restaurant in the north. Would you like to book a table?
AuGPT + HKS	B_2: hotel-internet: yes, hotel-parking: yes, hotel-stars: 4, hotel-type: guesthouse D_2: German, **Italian**, Spanish and English are **spoken** at Carolina Bed and **Breakfast**. R_2: **Yes**, they offer both Italian and English breakfast. Is there anything else I can help you with?
HyKnow (Single)	$\widetilde{B}_2$: hotel-internet: yes, hotel-parking: yes, hotel-stars: 4, hotel-type: guesthouse, **hotel-ruk: guesthouse**, **topic: breakfast option** D_2: **Breakfast** served at **Bridge Guest House** is Continental and full English/Irish. R_2: **No**, the hotel does not provide an Italian breakfast. Anything else I can do for you?
SeKnow-S2S (Single)	$\widetilde{B}_2$: hotel-internet: yes, hotel-parking: yes, hotel-stars: 4, hotel-type: guesthouse, **hotel-ruk: guesthouse**, **topic: breakfast option** D_2: Continental, vegetarian, and a full English/Irish **breakfast** are available at **Acorn Guest House**. R_2: **No**, they have a full English/Irish breakfast. Would you like to make a reservation?
SeKnow-PLM	$\widetilde{B}_2$: hotel-internet: yes, hotel-parking: yes, hotel-stars: 4, hotel-type: guesthouse, **hotel-ruk: Acorn Guest House**, **topic: breakfast option** D_2: Continental, vegetarian, and a full English/Irish **breakfast** are available at **Acorn Guest House**. R_2: **No**, they provide Continental, vegetarian, and a full English/Irish breakfast. Anything else I can do for you?

I. Case Study

In this section, we investigate a case of TOD when dialog steps into a newly inserted turn of modified MultiWOZ 2.1, involving unstructured knowledge. The example dialog segment (U_1, B_1, R_1, U_2) and corresponding output results of each model ($B_2/\widetilde{B}_2$, D_2, R_2) are presented in Table IX. Without utilization of unstructured knowledge, AuGPT misunderstands the user query, and recognizes the word "Italian" in U_2 as a constraint to update the belief state. As a result, the system makes an inappropriate recommendation. By combining with HKS, AuGPT predicts correct belief state, but fails in finding the relevant document, thus providing a wrong answer. This is because the wrong document's content has many common words with the dialog context, e.g. "Italian" and "Breakfast", which mislead the retrieval process. Although HyKnow (Single) accidentally gives an appropriate response, it gets interfered by the word "guesthouse" in R_1 and fails to predict the relevant entity in extended belief state, thus retrieving an incorrect document. In contrast, SeKnow-S2S (Single) uses the original triples in $\widetilde{B}_2$ to help locate the relevant entity, which avoids the document retrieval error caused by the vague prediction of *ruk*'s value. With the power of large-scale model pretraining, SeKnow-PLM does not get confused by the interference words in dialog context. Therefore, it successfully identifies the relevant entity ("Acorn Guest House") and topic ("breakfast option"), and generates a proper response with accurate information.

VII. CONCLUSION

In this paper, we define a task of modeling TOD with management of semi-structured knowledge. To address this task, we propose a TOD system SeKnow and introduce its two implementations, one (SeKnow-PLM) with and one (SeKnow-S2S) without model pretraining. Both implementations use an extended belief tracking to manage semi-structured knowledge, and jointly optimize TOD modeling grounded on structured and unstructured knowledge in the E2E manner. In the experiments, SeKnow shows strong performance in TOD modeling with semi-structured knowledge management, compared to existing TOD systems and their pipeline extensions. For future work, we consider evaluating our system on more various TOD scenarios where dialogs are grounded on semi-structured knowledge.

APPENDIX A
DOCUMENT PREPROCESSING

We preprocess the documents of each entity in the knowledge base of modified MultiWOZ 2.1 [15] dataset to extract the topic of each document, used as its retrieval index in the semi-structured knowledge management. Based on the TF-IDF [49] algorithm, we perform the topic word extraction domain-by-domain in a two-step procedure. First, we choose the top-three keywords with the highest TF-IDF scores in each document as its topic candidates. Then we filter the candidates to further select our desired topic words.

Noticing that different entities in the same domain usually have documents covering similar topics, we assume that a desired topic word should typically appear in multiple entities' documents, and therefore have a high frequency of occurrence among the topic candidates. So we calculate a cumulative average TF-IDF (CA-TF-IDF) score for each topic word in the candidates, which synthetically measures the word's document-level TF-IDF and entity-level occurrence frequency. Specifically, CA-TF-IDF sums the TF-IDF score of a topic word's each occurrence in the candidates, and divides it by the entity number in the domain. We filter out the topic candidates with low CA-TF-IDF scores and retain the rest to form the final retrieval indexes. The filtering thresholds are 2.3, 2.7, 6.9 and 7.3 for the domain of restaurant, hotel, taxi and train, respectively. While other domains are not involved in the unstructured documents. After the preprocessing, each document has one to three topic words extracted.

Appendix B
Training and Implementation Details

In SeKnow-S2S, we use two-layer Universal Transformers [14] to implement our encoders/decoders, with 8 parallel attention heads in each self-attention network. We set the convolution kernel size and inner-layer size of feed-forward network as 3 and 2048, and the batch size, embedding/hidden size and vocabulary size as 64, 512 and 3000, respectively. We also set dropout rate as 0.1 and use greedy decoding to generate the belief state and system response. Moreover, we use Adam optimizer [57] with an initial learning rate of $3e^{-4}$, and decay the learning rate by dividing it by current epoch number. We set the total number of training epoch as 15, with average training time 25 minutes per epoch using 1 GPU. In SeKnow-PLM, we follow AuGPT [7] to pretrain a GPT-2 [12] as our model backbone, using its suggested implementation settings[4]. We further finetune the GPT-2 on the training set of modified MultiWOZ 2.1, where we set the batch size as 8. We set the total number of finetuning epoch as 7, with average training time 100 minutes per epoch using 4 GPUs. Model training is performed on NVIDIA TITAN-Xp GPU. Besides, in semi-structured knowledge operation, we conduct the matching of entity name/ID and topic by using the fuzzy string matching toolkit[5].

Appendix C
Statistics of Modified MultiWOZ 2.1

There are totally 8449/1001/1004 dialogs[6] in the training, development and testing set of modified MultiWOZ 2.1, where 6501/836/847 dialogs have new turns inserted, respectively. After the modification, each dialog has 8.93 turns on average, which is longer than the original 6.85. The ontology of modified MultiWOZ 2.1 is same as the original, with 32 slot types (excluding *ruk*) and 2426 corresponding slot values. There are totally 291 entities in the semi-structured knowledge base: 110, 79, 66 and 33 entities in the domain of restaurant, attraction, hospital and hotel, respectively, and 1 entity in each of the other three domains[7] (i.e. police, train and taxi). Entities in the domain of restaurant, hotel, train and taxi are associated with unstructured documents, whose total number is 2882.

Acknowledgment

This work was supported by the NSFC projects (Key project with No. 61936010 and regular project with No. 61876096).

[4]https://huggingface.co/jkulhanek/augpt-bigdata/tree/main

[5]https://github.com/seatgeek/fuzzywuzzy

[6]These are slightly more compared to the original MultiWOZ 2.1, because some of the original dialogs are modified twice with different turns inserted.

[7]In the domain of train or taxi, we assume that different train schedules or taxi cars belong to a common entity.

References

[1] S. Gao, R. Takanobu, W. Peng, Q. Liu, and M. Huang, "Hyknow: End-to-end task-oriented dialog modeling with hybrid knowledge management," *arXiv preprint arXiv:2105.06041*, 2021.

[2] Y. Zhang, Z. Ou, and Z. Yu, "Task-oriented dialog systems that consider multiple appropriate responses under the same context," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 9604–9611.

[3] H. Le, D. Sahoo, C. Liu, N. Chen, and S. C. Hoi, "Uniconv: A unified conversational neural architecture for multi-domain task-oriented dialogues," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 1860–1877.

[4] Y. Zhang, Z. Ou, M. Hu, and J. Feng, "A probabilistic end-to-end task-oriented dialog model with latent belief states towards semi-supervised learning," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 9207–9219.

[5] E. Hosseini-Asl, B. McCann, C.-S. Wu, S. Yavuz, and R. Socher, "A simple language model for task-oriented dialogue," in *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[6] B. Peng, C. Li, J. Li, S. Shayandeh, L. Liden, and J. Gao, "Soloist: Few-shot task-oriented dialog with a single pre-trained auto-regressive model," *arXiv preprint arXiv:2005.05298*, 2020.

[7] J. Kulhánek, V. Hudeček, T. Nekvinda, and O. Dušek, "Augpt: Dialogue with pre-trained language models and data augmentation," *arXiv preprint arXiv:2102.05126*, 2021.

[8] J. Li, Q. Zhu, L. Luo, L. Liden, K. Huang, S. Shayandeh, R. Liang, B. Peng, Z. Zhang, S. Shukla *et al.*, "Multi-domain task completion dialog challenge ii at dstc9," in *Proceedings of the 35th AAAI Conference on Artificial Intelligence, Ninth Dialog System Technology Challenge Workshop*, 2021.

[9] S. Feng, H. Wan, C. Gunasekara, S. Patel, S. Joshi, and L. Lastras, "Doc2dial: A goal-oriented document-grounded dialogue dataset," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 8118–8128.

[10] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in neural information processing systems*, vol. 27, pp. 3104–3112, 2014.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[12] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[13] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, 2014, pp. 103–111.

[14] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and L. Kaiser, "Universal transformers," in *International Conference on Learning Representations*, 2018.

[15] S. Kim, M. Eric, K. Gopalakrishnan, B. Hedayatnia, Y. Liu, and D. Hakkani-Tur, "Beyond domain apis: Task-oriented conversational modeling with unstructured knowledge access," in *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2020, pp. 278–289.

[16] J. D. Williams, A. Raux, D. Ramachandran, and A. W. Black, "The dialog state tracking challenge," in *Proceedings of the SIGDIAL 2013 Conference*, 2013, pp. 404–413.

[17] M. Henderson, B. Thomson, and J. D. Williams, "The second dialog state tracking challenge," in *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*, 2014, pp. 263–272.

[18] L. El Asri, H. Schulz, S. K. Sarma, J. Zumer, J. Harris, E. Fine, R. Mehrotra, and K. Suleman, "Frames: a corpus for adding memory to goal-oriented dialogue systems," in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 2017, pp. 207–219.

[19] P. Budzianowski, T.-H. Wen, B.-H. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gasic, "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*, 2018, pp. 5016–5026.

[20] A. Rastogi, X. Zang, S. Sunkara, R. Gupta, and P. Khaitan, "Towards scalable multi-domain conversational agents: The schema-guided dia-

logue dataset," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020.
[21] Q. Zhu, K. Huang, Z. Zhang, X. Zhu, and M. Huang, "Crosswoz: A large-scale chinese cross-domain task-oriented dialogue dataset," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 281–295, 2020.
[22] T. Wen, D. Vandyke, N. Mrkšíc, M. Gašíc, L. Rojas-Barahona, P. Su, S. Ultes, and S. Young, "A network-based end-to-end trainable task-oriented dialogue system," in *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017-Proceedings of Conference*, vol. 1, 2017, pp. 438–449.
[23] S. Mehri, T. Srinivasan, and M. Eskenazi, "Structured fusion networks for dialog," in *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, 2019, pp. 165–177.
[24] W. Lei, X. Jin, M.-Y. Kan, Z. Ren, X. He, and D. Yin, "Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 1437–1447.
[25] W. Liang, Y. Tian, C. Chen, and Z. Yu, "Moss: End-to-end dialog system framework with modular supervision." in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 8327–8335.
[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 6000–6010.
[27] T. Zhao, A. Lu, K. Lee, and M. Eskenazi, "Generative encoder-decoder models for task-oriented spoken dialog systems with chatting capability," in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 2017, pp. 27–36.
[28] Z. Yu, A. W. Black, and A. I. Rudnicky, "Learning conversational systems that interleave task and non-task content," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017, pp. 4214–4220.
[29] S. Akasaki and N. Kaji, "Chat detection in an intelligent assistant: Combining task-oriented and non-task-oriented spoken dialogue systems," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1308–1319.
[30] L. Liao, R. Takanobu, Y. Ma, X. Yang, M. Huang, and T.-S. Chua, "Deep conversational recommender in travel," *arXiv preprint arXiv:1907.00710*, 2019.
[31] S. Yang, R. Zhang, and S. Erfani, "Graphdialog: Integrating graph knowledge into end-to-end task-oriented dialogue systems," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 1878–1888.
[32] T. Yu, R. Zhang, H. Er, S. Li, E. Xue, B. Pang, X. V. Lin, Y. C. Tan, T. Shi, Z. Li *et al.*, "Cosql: A conversational text-to-sql challenge towards cross-domain natural language interfaces to databases," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 1962–1979.
[33] S. Kim, M. Eric, B. Hedayatnia, K. Gopalakrishnan, Y. Liu, C.-W. Huang, and D. Hakkani-tur, "Beyond domain apis: Task-oriented conversational modeling with unstructured knowledge access track in dstc9," in *Proceedings of the 35th AAAI Conference on Artificial Intelligence, Ninth Dialog System Technology Challenge Workshop*, 2021.
[34] C.-H. Tan, X. Yang, Z. Zheng, T. Li, Y. Feng, J.-C. Gu, Q. Liu, D. Liu, Z.-H. Ling, and X. Zhu, "Learning to retrieve entity-aware knowledge and generate responses with copy mechanism for task-oriented dialogue systems," *arXiv preprint arXiv:2012.11937*, 2020.
[35] H. He, H. Lu, S. Bao, F. Wang, H. Wu, Z. Niu, and H. Wang, "Learning to select external knowledge with multi-scale negative sampling," *arXiv preprint arXiv:2102.02096*, 2021.
[36] D. Thulke, N. Daheim, C. Dugast, and H. Ney, "Efficient retrieval augmented generation from unstructured knowledge for task-oriented dialog," *arXiv preprint arXiv:2102.04643*, 2021.
[37] J. Kim, D. Ham, J.-G. Lee, and K.-E. Kim, "End-to-end document-grounded conversation with encoder-decoder pre-trained language model," 2021.
[38] T.-H. Wen, M. Gasic, N. Mrkšić, P.-H. Su, D. Vandyke, and S. Young, "Semantically conditioned lstm-based natural language generation for spoken dialogue systems," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1711–1721.
[39] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.
[40] J. Gu, Z. Lu, H. Li, and V. O. Li, "Incorporating copying mechanism in sequence-to-sequence learning," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 1631–1640.
[41] B. Byrne, K. Krishnamoorthi, C. Sankar, A. Neelakantan, B. Goodrich, D. Duckworth, S. Yavuz, A. Dubey, K.-Y. Kim, and A. Cedilnik, "Taskmaster-1: Toward a realistic and diverse dialog dataset," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 4506–4517.
[42] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 86–96.
[43] S. Edunov, M. Ott, M. Auli, and D. Grangier, "Understanding back-translation at scale," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 489–500.
[44] C. Federmann, O. Elachqar, and C. Quirk, "Multilingual whispers: Generating paraphrases with translation," in *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, 2019, pp. 17–26.
[45] C.-S. Wu, A. Madotto, E. Hosseini-Asl, C. Xiong, R. Socher, and P. Fung, "Transferable multi-domain state generator for task-oriented dialogue systems," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 808–819.
[46] M. Heck, C. van Niekerk, N. Lubis, C. Geishauser, H.-C. Lin, M. Moresi, and M. Gasic, "Trippy: A triple copy strategy for value independent neural dialog state tracking," in *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2020, pp. 35–44.
[47] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.
[48] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, pp. 1–67, 2020.
[49] C. D. Manning, H. Schütze, and P. Raghavan, *Introduction to information retrieval*. Cambridge university press, 2008.
[50] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: Bm25 and beyond," *Foundations and Trends® in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009.
[51] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880.
[52] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang, "Realm: Retrieval-augmented language model pre-training," *arXiv preprint arXiv:2002.08909*, 2020.
[53] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.
[54] S. Banerjee and A. Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, pp. 65–72.
[55] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.
[56] M. Eric, R. Goel, S. Paul, A. Sethi, S. Agarwal, S. Gao, A. Kumar, A. Goyal, P. Ku, and D. Hakkani-Tur, "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*, 2020, pp. 422–428.
[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.