KETOD: Knowledge-Enriched Task-Oriented Dialogue

Anonymous ACL submission

Abstract

Existing studies in dialogue system research mostly treat task-oriented dialogue and chit-chat as separate domains. Towards building a human-like assistant that can converse naturally and seamlessly with users, it is important to build a dialogue system that conducts both types of conversations effectively. In this work, we investigate how task-oriented dialogue and knowledge-grounded chit-chat can be effectively integrated into a single model. To this end, we create a new dataset, KETOD (Knowledge-Enriched Task-Oriented Dialogue), where we naturally enrich task-oriented dialogues with chit-chat based on relevant entity knowledge. We also propose two new models, SimpleToDPlus and Combiner, for the proposed task. Experimental results on both automatic and human evaluations show that the proposed methods can significantly improve the performance in knowledge-enriched response generation while maintaining a competitive task-oriented dialog performance. We believe our new dataset will be a valuable resource for future studies. The code and the dataset will be made publicly available.

1 Introduction

With the surge of deep neural networks and language model pre-training these years, there have been significant improvements in building conversational artificial intelligence. One major type of dialogue being studied is task-oriented dialogue (TOD) (Wen et al., 2017a; Budzianowski et al., 2018; Rastogi et al., 2020; Hosseini-Asl et al., 2020a), where the system aims to collect user intents/goals to complete certain tasks. In the current TOD systems, the system responses are mostly concise and templated, as we generally focus on the success of task completion but not providing a natural and engaging conversational experience. The latter is the target of another kind of popularly studied dialogue - knowledge-grounded chit-chat (Ghazvininejad et al., 2018; Zhang et al., 2018; Tuan et al., 2019; Dinan et al., 2019). In addition to simple casual chit-chat, knowledge-grounded chit-chat further enables the conversation around certain background knowledge, relevant subjects or world knowledge.

Existing studies mostly focus on one specific type of dialogue and have gained remarkable progress over the last few years. However, the ultimate goal of Conversational AI is a human-like, unified system capable of conversing with the users naturally and seamlessly among all kinds of dialogues. Current TOD systems can hardly make interesting and engaging conversations only with templated functional responses. Few previous works like ACCENTOR (Sun et al., 2021) have studied the combination of TOD and chit-chat, but their chit-chat augmentation is largely limited to simple general responses like ‘you’re welcome’, ‘sounds good to me’. In this work, we propose to enrich TOD with knowledge-grounded chit-chat, as one
step further towards the ultimate goal of building a human-like, unified system (See Figure 1 for an example). We believe that the proposed knowledge-enriched TOD system can conduct more social, natural, and engaging conversations.

To this end, we propose a new dataset, KETOD (Knowledge-Enriched Task-Oriented Dialogue). In order to obtain natural and high-quality knowledge-grounded chit-chat, we design the dataset construction framework by augmenting existing TODs and using the relevant entity knowledge to make the chit-chat enrichment. Specifically, for a given task-oriented dialogue, 1) extracting the entities from the dialogue states and actions; 2) retrieving the knowledge associated with the entities from external knowledge sources; 3) asking the human annotators to enrich the system responses with chit-chat using the retrieved knowledge. We demonstrate that the knowledge-enriched dialogues constructed with the proposed framework are consistently preferred by human judges across all axes of engagingness, interestingness, knowledge, and humanness.

We propose two models for our task, and study the challenges and insights of our new dataset. The first model is an end-to-end language model that jointly learns and generates both the TOD results (dialogue states and actions) and the knowledge-enriched responses. The second model is a pipeline model that first generates the TOD results, then uses another response generation model to generate the knowledge-enriched responses. We run comprehensive experiments to demonstrate the improvement over the baselines, and show that our models can generate better knowledge-enriched responses while maintaining competitive performance on the TOD task.

To summarize, we make the following major contributions:

- We propose the task of combining TOD and knowledge-grounded chit-chat.

- We construct a new large-scale dataset, KETOD, with high-quality, manually annotated dialogue responses enriched with knowledge-grounded chit-chat. We will release the dataset upon acceptance of the paper.

- We propose two models for our dataset, and carry comprehensive experiments to study the challenges and insights. We believe our dataset should be a valuable resource for building a human-like conversational assistant.

2 Related Work

Task-oriented dialogue. As the major application in current industry dialogue systems, task-oriented dialogue (TOD) has been one of the most popular types of dialogue in the research community. There have been many works on building each component of the TOD system, such as dialogue state tracking, action prediction, and response generation (Wen et al., 2015, 2017b; Mrksic et al., 2017; Zhong et al., 2018; Eric et al., 2020; Liu et al., 2018; Peng et al., 2017; Serban et al., 2017; Zhou et al., 2017). Later works begin to investigate building end-to-end systems (Bordes et al., 2017; Liu et al., 2018, 2017; Xu et al., 2020). Following the trend of building unified pre-trained models in NLP research, most recent works on TOD also apply such language model pre-training style methods on building end-to-end systems (Hosseini-Asl et al., 2020a; Peng et al., 2020; Su et al., 2021). Such methods have achieved top performances on various datasets.

Popular datasets in TOD include the DSTC challenge series (Williams et al., 2016), MultiWOZ (Budzianowski et al., 2018), SGD (Rastogi et al., 2020), etc. As the primary goal of TOD is the successful completion of the functional tasks, the system responses mostly tend to be concise and templated. In this work, we believe it’s the ultimate goal of dialogue research to build a unified dialogue assistant to naturally and seamlessly converse with all kinds of dialogues. And we take the next step of combining TOD with knowledge-grounded chit-chat by enriching the functional responses with external knowledge.

Chit-chat dialogue. Another type of popular studied dialogue is chit-chat, with the goal of making a natural and engaging conversation. Apart from the ‘pure’ simple chit-chat that mostly covers plain and general responses, more works focus on knowledge groundings to achieve better specificity and engagingness, such as using user profiles (Zhang et al., 2018), social media contexts (Sordoni et al., 2015), or knowledge graphs (Tuan et al., 2019; Moon et al., 2019), etc. In this work, our enriched chit-chat is grounded on open-domain knowledge, similar as the WOW dataset (Dinan et al., 2019), where the system converses with the users about certain topics involving entity knowledge in an open-ended setting. In contrast, WOW specifically focuses on
knowledge-grounded chit-chat, while our dataset combines TOD and such chit-chat.

Combination of task-oriented dialogue and chit-chat. There are few works on combining TOD with chit-chat. ACCENTOR (Sun et al., 2021) proposes to combine TOD with chit-chat by prepending or appending chit-chat to the TOD responses. But their chit-chat is mostly general responses like 'sounds good!', 'you’re welcome'. In contrast, our work proposes to combine TOD with knowledge-grounded chit-chat. FusedChat (Young et al., 2021) proposes to insert chit-chat turns into TOD as well as re-writing TOD turns, but their chit-chat is still mostly general responses or based on commonsense knowledge, without much knowledge-groundings.

3 The KETOD Dataset

3.1 Dataset Construction

In this section, we describe our framework to construct the KETOD dataset. We start from existing TOD datasets and employ human annotators to augment the functional system responses with knowledge-grounded chit-chat. The proposed approach is demonstrated to give natural, contextual-relevant knowledge enrichment, and meanwhile easy to scale to different datasets. Figure 2 gives an overview of the dataset construction pipeline.

Data preparation. We build upon the SGD dataset (Rastogi et al., 2020), with task-oriented dialogues spanning 16 domains, such as Restaurant, Weather, etc. Given each TOD, to obtain the knowledge relevant to the dialogue context, we first extract all the entities from the dialogue states and actions. We exclude the domains Alarm, Banks, and Payment as there are mostly no entities involved in these domains; Also, to simplify the human annotation process in the next step, we remove the dialogues with over 10 entities involved.

Knowledge retrieval. For each entity, we use the concatenation of the domain name and entity name as the query to retrieve Wikipedia articles. We use the DrQA retriever (Chen et al., 2017) to retrieve the top 2 Wikipedia articles and take the first 2 paragraphs of each article as the knowledge candidates associated with each entity. Then we break the retrieved articles into sentences, with each sentence as one knowledge snippet.

Response enrichment. In this step, we employ human annotators\(^1\) to enrich the system responses in the original TOD based on the dialogue context and the retrieved knowledge. For each TOD, we present to the annotators the full dialogue, as well as all the knowledge snippets associated with the entities in the dialogue. The annotators can click on each entity name to see the associated knowledge snippets in an expanded textbox. See Appendix A for our annotation interface.

The annotation process is as follows: 1) Read the full dialog first to have an overall story in mind, as well as the relevant knowledge snippets, then to decide how many turns to enrich with chit-chat and which turn(s) to enrich: If there is no way to make a natural chit-chat enrichment, skip the example. 2) After deciding the turn(s) to enrich with the chit-chat, select the knowledge snippets used to make the enrichment (at most 3 snippets for each turn); 3) Rewrite the system response to enrich with chit-chat grounded on the selected knowledge snippets; The functional information in the original response should be maintained, while may be rephrased to make the enriched response more natural.

To ensure the dataset quality, we first interview the annotators to select the appropriate hires through a few test examples. Then we launch a training session for all the annotators to learn the

\(^1\)Our annotators were hired as full-time employees through a leading annotation services vendor, and were paid in accordance with a fair wage rate.
Table 1: General statistics of KETOD.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Dialogues</td>
<td>5,324</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>27k</td>
</tr>
<tr>
<td>All turns</td>
<td>52,063</td>
</tr>
<tr>
<td>Turns enriched with chit-chat</td>
<td>6,302</td>
</tr>
<tr>
<td>All entities</td>
<td>4,639</td>
</tr>
<tr>
<td>All knowledge snippets</td>
<td>33,761</td>
</tr>
<tr>
<td>Avg. # turns per dialogue</td>
<td>9.78</td>
</tr>
<tr>
<td>Avg. # tokens in enriched responses</td>
<td>28.07</td>
</tr>
<tr>
<td>Avg. # entities per dialogue</td>
<td>4.98</td>
</tr>
<tr>
<td>Avg. # knowledge snippets per dialogue</td>
<td>70.50</td>
</tr>
</tbody>
</table>

Table 1: General statistics of KETOD.

task and the annotation interface. We launch the official batches after the annotators can well-master the task. During annotation, we specifically emphasize the contextualization of the knowledge-grounded chit-chat - the enrichment should be contextualized closely on the dialogue context, but not a plain restatement of the knowledge snippets.

3.2 Dataset Statistics and Analysis

We end up with 5,324 dialogues with enriched system responses. We make the split of 4,247/545/532 as the train/dev/test set. Table 1 shows the statistics of the KETOD dataset. Around 12.1% of the turns (which indicates mostly 1 or 2 turns in one dialogue) are enriched with knowledge-grounded chit-chat. This intuitively complies with our goal of making the whole dialogue natural and engaging, since too frequent chit-chat may result in redundancy and unnaturalness.

Quality assessment of the annotation. During the annotation process, around 12% of the dialogues cannot be enriched with any turns and thus discarded. To assess the quality of the annotation, we sample 5% of the annotated dialogues and distribute them to linguistics to check: 1) If the chit-chat enrichment is relevant and natural; 2) If the knowledge snippets are accurately selected corresponding to the enrichment. We end up with a correct rate of 87.0%.

Justification of the chit-chat enrichment. To demonstrate that our proposed knowledge-enriched TOD can be more natural and engaging, we conduct human evaluations to compare KETOD dialogues and their corresponding original TOD dialogues without chit-chat enrichment (SGD). We follow (Li et al., 2019) to make pairwise comparisons of the full dialogues over the following four axes: engagingness, interestingness, knowledge, and humanness. The results in Figure 3 show the superiority of KETOD over all axes.

4 Approaches

In this section, we will describe the proposed two models for the KETOD dataset.

4.1 Overview and Formulations

For each dialogue turn, denote the dialogue context (history) as $C$, belief states as $B$, database search results as $D$, actions as $A$, the knowledge snippets used for chit-chat enrichment as $K$, the response as $T$. Then we formulate the problem as: given the dialogue context $C$ and a knowledge source (Wikipedia in this dataset), the target is to generate the belief states $B$, actions $A$, and the response $T$, which may be enriched with chit-chat grounded on the knowledge based on the context. The goal of the optimization on KETOD is two-folded: 1) Optimizing the generation of knowledge-enriched responses; 2) Maintaining the task performances;

In this work, we propose the following modeling framework on KETOD: 1) given the dialogue context, generate the belief states and actions; 2) extract the entities in the belief states and actions, then use these entities to retrieve knowledge candidates (similar as in the dataset construction process); 3) conditioned on the dialogue context, use a knowledge selection model to select knowledge snippets from the knowledge candidates retrieved; 4) generate the knowledge-enriched response conditioned on both the dialogue context and the selected knowledge snippets.

Based on the above general framework, we propose two architectural approaches, SimpleToD-Plus and Combiner, respectively in §4.3 and §4.4.

4.2 Knowledge Selection

After the generation of belief states and actions, we retrieve the knowledge snippet candidates from Wikipedia using the entities in the belief states
and actions. The average number of knowledge snippets candidates retrieved for each dialogue is around 70. It is impractical to input all of them into the models due to the large amount. As we have the annotation for the ground truth knowledge snippets used for each chit-chat enrichment, we train a knowledge selection model to select the top knowledge snippets most appropriate for chit-chat enrichment. Specifically, we concatenate the dialogue context with each knowledge snippet as the input. Then we use BERT (Devlin et al., 2019) to train a simple classifier to rank all the knowledge snippets candidates. We take the top 3 ones as the knowledge selection results. We use the same knowledge selection model for both architectures.

4.3 SimpleToDPlus

SimpleToD (Hosseini-Asl et al., 2020b) is a recent popular approach on TOD, which uses one single language model to sequentially generate the belief states, actions, and responses. It has achieved strong performances in all the above functional tasks. In this work, we propose its extension, SimpleToDPlus, to generate knowledge-enriched responses for TOD. The left part of Figure 4 shows the overview of SimpleToDPlus.

We formulate the training sequence as:

\[ [C, B, D, A, K, <\text{chitchat}>, T] \] (1)

Where \(<\text{chitchat}\>\) is a tag to indicate the decision of whether to enrich the response with knowledge grounded chit-chat or not. If the response is not enriched, we insert the tag \(<\text{nochitchat}\>\). Since the number of the gold knowledge snippets varies from 1 to 3 (as in the dataset construction), to be compatible with inference time, here we first run the knowledge selection model on all training instances. Then we construct the knowledge snippets \(K\) as the merge of the gold knowledge snippets and the knowledge selection model results, truncated to 3 ones. If the response is not enriched with chit-chat, i.e., no gold knowledge snippets, we still put 3 snippets from the knowledge selection model ranking results here during training.

In the inference time, we first sequentially generate the belief states and actions. Then we extract the entities from the generated belief states and actions, and apply the same process of knowledge retrieval as in dataset construction. Next, we run the knowledge selection model on the retrieved knowledge candidates and take the top 3 knowledge snippets as the model input followed by the generated actions. At last, the model generates the decision to make chit-chat enrichment or not, followed by the final response.

Since the knowledge-enriched response is conditioned on the entity knowledge from the belief states and actions, we need to directly include the entities in the actions and responses during generation, instead of generating a delexicalized result first and then doing lexicalizing in the post-process as in the original SimpleToD. To simplify, we use the oracle database search results for all the experiments in this work.

4.4 Combiner

SimpleToDPlus models all the generations in an end-to-end manner. In Combiner, we use a pipeline of a TOD model followed by a response generation model to separate the TOD part (belief states, actions) with the generation of knowledge-enriched responses. The goal is to study whether an independent model can better learn each task with less interference from the other. The overview of the architecture is shown on the right of Figure 4.

For the TOD model, we use SimpleToD to generate the belief states and actions, with the training sequence as:

\[ [C, B, D, A] \] (2)

In the experiments, we find that including the knowledge-enriched responses here during training degrades the task performance, indicating the disturbance from the ungrounded knowledge in the responses.

Figure 4: Illustration of the models. Left: the SimpleToDPlus model; Right: the Combiner model.
For the response generation model, we use GPT-2 (Radford et al., 2019) with the concatenation of the dialogue context, actions, and the knowledge snippets as the prompt:

\[ T = \text{GPT-2}(C, A, K) \]  

(3)

We use the same way of constructing the merged knowledge snippets during training, and the same process of knowledge retrieval and selection during inference as in SimpleToDPlus.

5 Experimental Results

Baseline model. We use the original SimpleToD (Hosseini-Asl et al., 2020b) as our baseline model, i.e., with the training sequence as \([C, B, D, A, T]\) without the injection of knowledge snippets. Therefore here, the knowledge-grounded chit-chat in the responses \(T\) do not have any knowledge groundings - we aim to show the necessity of knowledge grounding for our task, as well as the effectiveness of our proposed models to incorporate knowledge.

Experimental setups and evaluations. For all models, we use the Adam optimizer (Kingma and Ba, 2015). Check Appendix B for more details of training and parameter settings. For the TOD performances, we evaluate the belief states with joint goal accuracy (Joint GA) and average goal accuracy (Avg GA), and the actions with act-slot F1, same as (Sun et al., 2021). For the automatic evaluations of response generation, we use three BLEU-4 scores: BLEU-4\textsubscript{aug} for evaluating the responses enriched with knowledge; BLEU-4\textsubscript{orig} for evaluating the responses not enriched with knowledge; BLEU-4\textsubscript{all} for evaluating all responses;

5.1 Main Results

Performance on response generation. Table 2 shows our main experiment results. For the performances on response generation, we can see that both of our proposed models, SimpleToDPlus and Combiner, improve on the knowledge-enriched response generation (BLEU-4\textsubscript{aug}) over the SimpleToD baseline. Since in the baseline, we do not include the knowledge snippets in the input, the generated responses are mostly enriched with random knowledge or frequent knowledge in the training data. The improvements demonstrate the necessity of knowledge grounding and the effectiveness of the proposed knowledge enrichment methods. Combiner performs slightly better on knowledge-enriched responses than SimpleToDPlus but falls short on the responses without knowledge-enrichment (i.e., original TOD responses). This is partially because of its pipeline nature - a separated response generation module can better learn the knowledge enrichment without the disturbance of other tasks, but the error cascading from the generated actions degrades the performance of the TOD responses part.

Performances on belief states and actions. To better study how the knowledge enrichment affects the TOD performances, we first train SimpleToD on our dataset without the knowledge enrichment, i.e., replace all the knowledge-enriched responses with the original responses in SGD. We name it as SimpleToD-ref in Table 2, serving as a reference of the original TOD performances. The SimpleToD baseline gives largely degraded performances due to the disturbance from the ungrounded knowledge in the responses during training. Therefore in Combiner, we do not include the responses in the training sequences of the TOD model (specified in section 4.4), and obtain better scores. SimpleToDPlus achieves the best TOD performances, which are nearly competitive with SimpleToD-ref.

Human evaluations. In order to get the more comprehensive measure of the response generation performances, we conduct human evaluations for both dialogue-level pairwise comparison and turn-level factuality evaluation. For dialogue-level pairwise comparison, we randomly sample 200 dialogues from the test set and apply the same process as in dataset evaluation (3.2). For each model, we construct the full dialogue results by concatenat-

<table>
<thead>
<tr>
<th>Models</th>
<th>Joint GA</th>
<th>Avg GA</th>
<th>Act-Slot F1</th>
<th>BLEU-4\textsubscript{aug}</th>
<th>BLEU-4\textsubscript{orig}</th>
<th>BLEU-4\textsubscript{all}</th>
</tr>
</thead>
<tbody>
<tr>
<td>SimpleToD-ref</td>
<td>27.6</td>
<td>54.2</td>
<td>67.6</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>SimpleToD</td>
<td>23.7</td>
<td>50.1</td>
<td>62.7</td>
<td>4.8</td>
<td>10.7</td>
<td>10.0</td>
</tr>
<tr>
<td>SimpleToDPlus</td>
<td>28.6</td>
<td>52.2</td>
<td>66.9</td>
<td>6.3</td>
<td>11.7</td>
<td>11.0</td>
</tr>
<tr>
<td>Combiner</td>
<td>24.5</td>
<td>51.5</td>
<td>64.5</td>
<td>6.5</td>
<td>9.9</td>
<td>9.5</td>
</tr>
</tbody>
</table>

Table 2: Main experiment results: Both SimpleToDPlus and Combiner outperform the baseline. Overall SimpleToDPlus obtains better response generation performance while maintaining competitive TOD performance.
There are several inference stages for this task—the TOD results (belief states and actions), the selection of knowledge snippets, and the final response generation. Where each stage is conditioned on previous results. Therefore, the errors accumulate through all the stages leading to the final performances. Here we run another two sets of experiments to study such error accumulations and compare the two models. Specifically, first, we feed the models with the gold TOD results, chit-chat decisions, and knowledge snippets, to solely test the abilities to generate the knowledge-enriched responses. Second, we feed the models with the gold TOD results to test the following stages of knowledge selection and the response generation. The results are shown in Table 5. Compared with the full inference results in Table 2, we can see that the Combiner model largely outperforms SimpleToDPlus if provided with more gold results for previous stages. However, it gradually falls behind SimpleToDPlus when moving to fully end-to-end inference due to the error cascading of its pipeline nature.

**Importance of knowledge selection strategies.** To demonstrate the importance of the knowledge selection strategies (and their subsequent recall performance), we run SimpleToDPlus with 1) gold knowledge snippets; 2) predicted knowledge snippets (with BERT); 3) knowledge snippets selected by heuristics (we use TF-IDF matching between pets (with BERT); 2) knowledge snippets selected by heuristics (we use TF-IDF matching between

<table>
<thead>
<tr>
<th>Metrics</th>
<th>SimpleToDPlus win (%)</th>
<th>Combiner win (%)</th>
<th>Tied (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engagingness</td>
<td>47.8</td>
<td>24.5</td>
<td>27.8</td>
</tr>
<tr>
<td>Interestingness</td>
<td>34.5</td>
<td>19.0</td>
<td>46.5</td>
</tr>
<tr>
<td>Knowledge</td>
<td>29.5</td>
<td>26.3</td>
<td>44.3</td>
</tr>
<tr>
<td>Humanness</td>
<td>43.3</td>
<td>23.8</td>
<td>33.0</td>
</tr>
</tbody>
</table>

Table 3: Human evaluation of SimpleToDPlus vs. Combiner.

<table>
<thead>
<tr>
<th>Metrics</th>
<th>SimpleToDPlus win (%)</th>
<th>Gold win (%)</th>
<th>Tied (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engagingness</td>
<td>16.8</td>
<td>60.5</td>
<td>22.8</td>
</tr>
<tr>
<td>Interestingness</td>
<td>12.0</td>
<td>51.0</td>
<td>37.0</td>
</tr>
<tr>
<td>Knowledge</td>
<td>14.5</td>
<td>44.8</td>
<td>40.8</td>
</tr>
<tr>
<td>Humanness</td>
<td>17.3</td>
<td>58.0</td>
<td>24.8</td>
</tr>
</tbody>
</table>

Table 4: Human evaluation of SimpleToDPlus vs. Gold.

5.2 Ablations and Analysis

**Analysis of different inference stages.** There are several inference stages for this task—the TOD results (belief states and actions), the selection of knowledge snippets, and the final response generation, where each stage is conditioned on previous results. Therefore, the errors accumulate through all the stages leading to the final performances. Here we run another two sets of experiments to study such error accumulations and compare the two models. Specifically, first, we feed the models with the gold TOD results, chit-chat decisions, and knowledge snippets, to solely test the abilities to generate the knowledge-enriched responses. Second, we feed the models with the gold TOD results to test the following stages of knowledge selection and the response generation. The results are shown in Table 5. Compared with the full inference results in Table 2, we can see that the Combiner model largely outperforms SimpleToDPlus if provided with more gold results for previous stages. However, it gradually falls behind SimpleToDPlus when moving to fully end-to-end inference due to the error cascading of its pipeline nature.

**Importance of knowledge selection strategies.** To demonstrate the importance of the knowledge selection strategies (and their subsequent recall performance), we run SimpleToDPlus with 1) gold knowledge snippets; 2) predicted knowledge snippets (with BERT); 3) knowledge snippets selected by heuristics (we use TF-IDF matching between the current dialogue turn and the knowledge snippets). To eliminate the influences brought by other inference stages, we feed the model with gold TOD results (dialogue states and actions). The results are shown in Table 6. There exists a certain level of variance for knowledge selection, e.g., when recommending a song for the user, you may talk about

<table>
<thead>
<tr>
<th>BLEU-4_{aug}</th>
<th>BLEU-4_{aug}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>9.7</td>
</tr>
<tr>
<td>BERT selection</td>
<td>7.8</td>
</tr>
<tr>
<td>TF-IDF selection</td>
<td>6.6</td>
</tr>
</tbody>
</table>

Table 5: Analysis of different inference stages: we provide the models with gold results up to certain stages, and investigate the performances for the inferences on following stages.

<table>
<thead>
<tr>
<th>Knowledge selection recall (%)</th>
<th>BLEU-4_{aug}</th>
<th>BLEU-4_{aug}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>9.7</td>
<td>14.6</td>
</tr>
<tr>
<td>BERT selection</td>
<td>7.8</td>
<td>14.4</td>
</tr>
<tr>
<td>TF-IDF selection</td>
<td>6.6</td>
<td>13.7</td>
</tr>
</tbody>
</table>

Table 6: SimpleToDPlus response generation performance with varying knowledge selection strategies.
Learning **when** to inject knowledge-enriched chit-chat. In all models, we use the special token `<chichat>` and `<nochichats>` to indicate the decision to inject knowledge enrichment for the responses. To study the effect of the chit-chat injection decision-making accuracy on the overall dialogue tasks, we run SimpleToDPlus (1) with the ground-truth information of turns to enrich with chit-chat, and (2) with the predicted decisions, using the gold TOD results. Table 7 shows the performance gap, which highlights the importance of knowing **when** to inject knowledge-enriched chit-chat. While such decisions are conditioned on the dialogue history, e.g., we may tend to not enrich a turn if many of the previous turns are enriched to avoid redundancy, there also exists some variance. In a real system, we may consider specifying the turns to make the chit-chat enrichment instead of letting the model make the decision.

### Domain analysis

We investigate the model performance for each domain in Table 8. We observe that the performance differences may depend on the variance of the enriched knowledge. Domains with larger variance on selected knowledge tend to have lower automatic scores. For example, in the **hotels** domain, mostly the chit-chat is about the locations since there are mostly location entities involved in this domain. But for the **restaurants** domain, the enriched knowledge can be about the food, the restaurant, as well as the location. The selected knowledge shows more diversity and variance.

We provide case studies in Figure 5 to compare the predicted results with the gold references.

### 6 Conclusion and Future Work

In this work, we propose to combine task-oriented dialogue with knowledge-grounded chit-chat as one step further towards building a unified, human-like conversational assistant. To this end, we construct a new dataset named KETOD, with manually composed knowledge-enriched system responses. We conduct comprehensive experiments on our new dataset to study the insights and challenges. In real-world conversations, there are many ways that different kinds of dialogues are fused together. In this work, we start from the more straightforward type of directly enriching the TOD responses with knowledge-grounded chit-chat. Potential future works could be on studying other fusion strategies, for example, interleaving TOD dialogue turns with chit-chat turns, chit-chat grounded on multiple types of knowledge sources, etc.
7 Ethical Considerations

Data Access and Licensing. We develop the KETOD dataset based on the publicly available SGD dataset\(^2\) (Rastogi et al., 2020). The SGD dataset is publicly available under the CC-BY-SA-4.0 License.

Dataset Collection Process and Conditions. For the annotation of our KETOD dataset, linguistics for assessing data quality, and all the human evaluations, our annotators were hired as full-time employees through a leading annotation services vendor, and were paid in accordance with a fair wage rate.

References


Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. Opendialkg: Explainable conversational reasoning with attention-based walks over


### Appendix A: Dataset Construction

Figure 6 shows our annotation interface to add knowledge-grounded chit-chat to TOD. The left part shows the full dialogue, where the annotators can click and expand each turn to make the chit-chat enrichment. The right part shows all the entities with the associated knowledge snippets. The annotators can click on each entity name to expand the textbox to see the knowledge snippets. We add index number to each knowledge snippet (shown in green brackets), and the annotators are asked to write down the indexes of the knowledge snippets they used for writing the knowledge grounded chit-chat. Figure 7 shows one example annotation turn using our interface.

### Appendix B: Model and Training Details

All the implementations are based on the Huggingface Transformers library. For the knowledge selection model, we use BERT-base with learning rate of 3e-5 and batch size of 16. For the baseline SimpleToD model, SimpleToDPlus model, and Combiner model, we all use learning rate of 1e-4 and batch size of 16. All the experiments are done using TESLA M40 GPU cards.

---

3https://github.com/huggingface/transformers
Figure 6: Our annotation interface example 1.

Figure 7: Our annotation interface example 2.