# Learn More from Less: Improving Conversational Recommender Systems via Contextual and Time-Aware Modeling

**Anonymous ACL submission**

## Abstract

Conversational Recommender Systems (CRS) aims to perform recommendations through interactive conversations. Prior work on CRS tends to incorporate more external knowledge to enhance performance. Given the fact that too much extra knowledge introduces the difficulty to balance among them and degenerates the generalizability, we propose to fully discover and extract the internal knowledge from the context. We capture both entity-level and contextual-level representations to jointly model user preferences for the recommendation, where a time-aware attention is designed to emphasize the recently appeared items in entity-level representations. We further use the pre-trained BART to initialize the generation module to alleviate the data scarcity and enhance the context modeling. Experiments on two public CRS datasets show that our model achieves comparable performance with less external knowledge and generalizes well to other domains. Further analyses demonstrate the effectiveness of our model in different scenarios.

## 1 Introduction

Conversational Recommender Systems or **CRS** (Li et al., 2018; Chen et al., 2019; Zhou et al., 2020a; Lu et al., 2021) have recently attracted many researchers due to the booming of e-commerce platforms. A CRS aims to provide high-quality recommendations to users through engaged conversations. Different from the traditional recommender systems, it focuses on learning users' preferences through natural language interaction with users, and has a high impact in e-commerce.

An effective CRS is expected to be able to clarify user intents, learn user preferences, recommend high-quality items and reply to users with suitable responses. Previous studies on CRS generally divide it into two parts: a *recommendation* module and a *generation* module. To improve the recommendation performance, previous efforts (Zhou et al., 2020a; Lu et al., 2021) focus on including more and more external knowledge into the system, as most of the available CRS datasets are relatively small (due to the expensive annotation process) (Li et al., 2018; Moon et al., 2019) and hard to extract meaningful features based on the context alone. For example, to improve the performance in conversational movie recommendation, entity-level knowledge graph (Chen et al., 2019), word-level knowledge graph (Zhou et al., 2020a) and item reviews (Lu et al., 2021) are successively introduced into the system.

However, there are three issues existing in the development of the previous methods. First, though the performance is improved by introducing more external knowledge, how to balance them in a single system becomes a new challenge. Second, the collection and annotation of the external knowledge needs much human effort. Third, the collected external knowledge may lack generalizability when facing broader application scenarios. On the other hand, for generation, most of the previous methods employ a general encoder-decoder framework and train the model from scratch. It also suffers from the overfitting issue on the relatively small dataset.

In this work, instead of exploring more external knowledge to assist the learning of user preferences, we choose to fully discover and extract the internal knowledge from the context. Concretely, we capture both entity-level and contextual-level representations to model the user preferences. The entity-level representations summarize user preferences with the appeared items in the context. Rather than applying a learnable matrix to let the model learn the importance of each item on its own (Chen et al., 2019; Lu et al., 2021), we design a new time-aware attention to organize the appeared items for emphasizing the more recently appeared items. As one of the goals of CRS is to guide the users to express their preferences explicitly through conversations, we believe that more recently appeared items can reflect users' interests better. As for the contextual-level representations extracted by a context encoder, they reflect the semantic- and discourse-level user preferences, which cannot be captured by entity-level ones. These two representations complement each other to enhance the final recommendation results. Besides, to alleviate the data scarcity effect in capturing meaningful context features, we use the pre-trained BART (Lewis et al., 2020) model to initialize our generation module.

We conduct experiments on two public and popular CRS datasets ReDial (Li et al., 2018) and OpenDialKG (Moon et al., 2019). The results show that our model can achieve high-quality and comparable performance when assisted with less external knowledge, and validate that our model generalizes well to other domains. Further analyses also demonstrate the effectiveness of our proposed methods in different scenarios.

The main contributions of this work can be summarized as follows:

- We propose to combine both entity-level and contextual-level representations for conversational recommendation, which achieves comparable performance with less external knowledge and generalizes well to other domains.
- We point out the limitation of the previous entity modeling method and design a time-aware attention to enhance entity modeling for recommendation.
- We examine the effects of BART pre-training and also conduct extensive experiments to show that our model is effective in different scenarios.

## 2 Related Work

Various task formulations with different hypotheses and application scenarios in CRS have been proposed in recent years. We summarize them into three categories and introduce them as below.

**Question Driven Systems.** As the rating or click feedback in traditional recommender systems is limited in that they do not exactly tell why users like or dislike an item, the feedback from the user may be very sparse. Question driven systems are proposed to effectively understand users' preference and improve the recommendations over time by asking clarifying questions. Christakopoulou et al. (2018) propose question-based video recommender system. Zhang et al. (2018) build systems based on aspect-centered questions. Aliannejadi et al. (2019) formulate the task of asking clarifying questions in open-domain information-seeking conversational systems. More recent works focus on asking attribute-central questions and develop reinforcement learning based approaches (Lei et al., 2020a; Ren et al., 2020; Deng et al., 2021) or graph based approaches (Xu et al., 2020; Lei et al., 2020b; Ren et al., 2021; Xu et al., 2021).

**Strategies Learning in Multi-turn CRS.** Some works focus on balancing the trade-off between exploration (i.e., asking questions) and exploitation (i.e., making recommendations), especially for cold-start users. They study the trade-off strategies to achieve engaging and successful recommendations. Some of them (Li et al., 2010, 2016; Christakopoulou et al., 2016; Li et al., 2020) leverage bandit online recommendation methods and focus on cold-start scenarios, while others work on strategically asking clarification questions with fewer turns (Lei et al., 2020a,b; Sun and Zhang, 2018).

**Open-ended CRS.** An open-ended CRS tends to make recommendation in a more natural and casual way compared with the task-oriented CRS. Many datasets have been collected or built to push forward the research of CRS, including ReDial (Li et al., 2018), TG-ReDial (Chinese) (Zhou et al., 2020b), GoRecDial (Kang et al., 2019), DuRecDial (Chinese) (Liu et al., 2020), INSPIRED (Hayati et al., 2020) and OpenDialKG (Moon et al., 2019) datasets. Most of them consist of around 10,000 conversations that are focused on recommendation and chit-chat on different domains. For example, ReDial is about movie recommendation, while OpenDialKG is concerned with several domains, including movie, book, sports and music. The follow-up studies based on the ReDial dataset generally divide the CRS into recommendation and generation modules. For the recommendation module, the previous works tend to apply more and more external knowledge to improve the recommendation performance, e.g., entity-level knowledge graph (Chen et al., 2019), word-level knowledge graph (Zhou et al., 2020a) and item reviews (Lu et al., 2021)). However, it's difficult to manage so much external knowledge via an end-to-end model. What's more, some (like item reviews) might need much effort to collect and annotate, and are not generally applicable for all kinds of domains (e.g., some items might lack reviews). For generation module, most of the previous works adopt encoder-decoder framework and train the generation model from scratch. However, it's difficult to learn diverse and valuable patterns from relatively small datasets. Our work further explores approaches for this category.

Different from the above methods, we are more interested in capturing better user representations from conversation context rather than adding more external knowledge and utilizing the pre-trained models (i.e., BART) to enhance our generation.

## 3 Methodology

In this section, we first formulate the task in §3.1, followed by our generation module in §3.2, which is finetuned with BART (Lewis et al., 2020). Then we introduce how we produce recommendation based on conversation context in §3.3. Finally in §3.4, we describe how we integrate the above two modules and produce the final responses.

### 3.1 Problem Formulation

A CRS generally consists of two modules, named as *generation* module and *recommendation* module. It takes a conversation context $C = (t_1, \dots, t_m)$ as input, where $t_i$ is an utterance from the seeker (i.e. the user) or the CRS itself and $m$ is the number of context utterances. The CRS uses its recommendation module to recommend items $\mathcal{I}_i$ from a candidate item set $\mathcal{I}$ and embeds them into a response $R = (y_1, \dots, y_n)$, a sequence of $n$ tokens generated by the generation module based on the conversation context $C$.

### 3.2 BART-based Response Generation

Our response generation module follows a general Transformer (Vaswani et al., 2017) sequence-to-sequence framework. As most of the available CRS datasets are relatively small and contain only around 10K conversations (Li et al., 2018; Moon et al., 2019), it is difficult to learn complex semantic and discourse level dependencies only based on the training corpus. To relieve the burden and enhance context modeling, we choose to finetune a pre-trained BART (Lewis et al., 2020) model for our response generation. BART was trained with several denoising objectives on large-scale books and Wikipedia data, and has been shown to be effective in many generation tasks including abstrastive QA, summarization, machine translation at the sentence and document level, and persona-based response generation.

Specifically, to enable BART to generate item-related responses, we extend its original vocabulary $\mathcal{V}$ with the item set $\mathcal{I}$ to be $\mathcal{V}' = \mathcal{V} \cup \mathcal{I}$, and use a CRS training corpus to finetune the model. During finetuning, we concatenate the utterances $t_i$ in context $C$ with an appended $\langle EOT \rangle$ token in their chronological order as the input, and maximize the probability of the ground truth response $R$. The whole process is summarized as follows:

$$\boldsymbol{H}^C = \text{Transformer\_Encoder}(w_C) \quad (1)$$

$$y_k = \text{Transformer\_Decoder}(y_{<k}, \boldsymbol{H}^C) \quad (2)$$

$$\mathcal{L}_{gen} = \sum\nolimits_{k=1}^{n} -\log(p(y_k | y_{<k}, \boldsymbol{H}^C)) \quad (3)$$

where $w_C = [t_1; \langle EOT \rangle; t_2; ..; \langle EOT \rangle; t_m]$, and $y_{<k}$ represents the target tokens before $y_k$.

An integration operation is added to the generation output to make the generation aware of the recommendation. We will discuss it later in §3.4.

### 3.3 Context-Time-Aware Recommendation

To fully understand user preferences over items from a given context $C$, we propose to extract two kinds of information for the recommendation. The first is entity-level information, where we extract the mentioned entities (including the items in the item set $\mathcal{I}$) from $C$ and apply them to an external related knowledge graph to perform entity linking (Daiber et al., 2013). The second is contextual information represented by BART representations $\boldsymbol{H}^C$, which is expected to capture information from the perspectives of semantic and conversational discourse. We describe the details in the following.

**Entity-level Representation.** We employ a relational knowledge graph (e.g. DBpedia) to enhance entity modeling. Specifically, we denote a triplet in the knowledge graph with $\langle e_1, r, e_2 \rangle$, where $e_1, e_2 \in \mathcal{E}$ are entities from the entity set $\mathcal{E}$ and $r$ is an entity relation from the relation set $\mathcal{R}$.

We use an R-GCN (Schlichtkrull et al., 2018) to encode relation-aware entity representations. Formally, the representation of an entity $e$ at the $(l+1)$-th layer is calculated as follows:

$$\boldsymbol{h}_e^{(l+1)} = \text{ReLU}\Big(\sum_{r \in \mathcal{R}'} \sum_{e' \in \mathcal{E}_e^r} \frac{1}{Z_{e,r}} \boldsymbol{W}_r^{(l)} \boldsymbol{h}_{e'}^{(l)}\Big) \quad (4)$$

where $\boldsymbol{h}_e^{(l)}$ is the representation of entity $e$ at the $l$-th layer, and $\mathcal{E}_e^r$ denotes the set of neighboring nodes of $e$ under the relation $r$; $\mathcal{R}' = \mathcal{R} \cup \{r_{self}\}$ contains all the relations including self loop; $\boldsymbol{W}_r^{(l)}$ is a learnable relation-specific transformation matrix and $Z_{e,r}$ is a normalization factor. For sim-

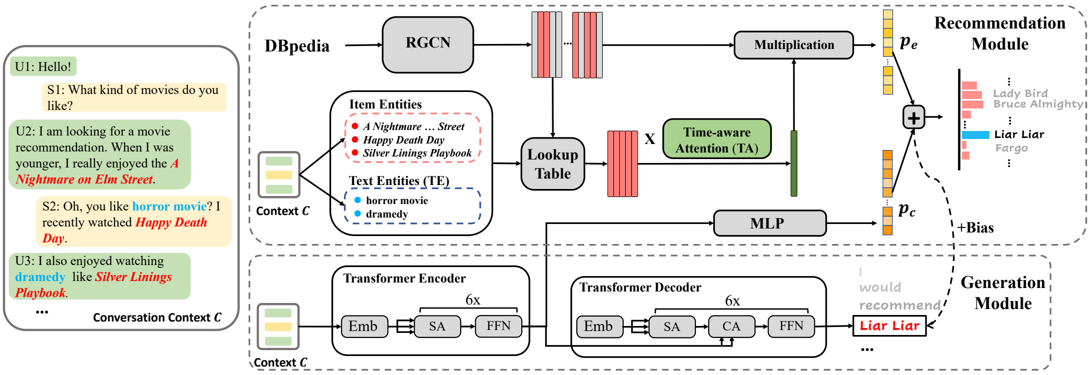


Figure 1: Our framework for conversational recommendation.

plicity, we represent the representations in the final layer $L$ as $\boldsymbol{h}_e$ by omitting the superscript "$L$".

Given a context $C$, we extract the entities as user preference $\mathcal{T}_u = e_1, e_2, ..., e_{|\mathcal{T}_u|}$ from two perspectives: item entities (i.e., entities that appear in $\mathcal{I}$) and other relevant contextual entities (mentioned in utterances but not an item in $\mathcal{I}$, e.g., an actor of a film item. We denote them as *text entities*). The entities $e_i \in \mathcal{E}$ are sorted in the order of appearance. After looking up the entities in $\mathcal{T}_u$ from $\boldsymbol{H} = \{\boldsymbol{h}_e\}_{e=1}^{|\mathcal{E}|}$ , we get $(\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_{|\mathcal{T}_u|})$.

To summarize the entity-level user representation, previous work mainly depends on the self-attention mechanism (Zhou et al., 2020a; Lu et al., 2021), where a learnable matrix is leveraged to learn and derive each entity's importance. Such a mechanism might be sub-optimal as no supervised signals are used to guide the model to learn knowledge about entity importance. Instead, we propose **Time-aware Attention**, where the entity-level user representation $\boldsymbol{h}^E$ is calculated as follows:

$$\boldsymbol{h}^E = \sum_{i=1}^{|\mathcal{T}_u|} \frac{\lambda^{i-1}}{\sum_{i=1}^{|\mathcal{T}_u|} \lambda^{i-1}} \boldsymbol{h}_i \tag{5}$$

where $\lambda \geq 1$ is a hyper-parameter to control the recency effect. This means that the recently appeared items will contribute more to the next item prediction, which is consistent with the intuition to such a system . Finally, the entity-level recommendation probability is computed as follows:

$$\boldsymbol{p}_e = \mathrm{softmax}(\mathrm{mask}(\boldsymbol{h}^{\boldsymbol{E}} \boldsymbol{H}^\top)) \tag{6}$$

where $\mathrm{mask}$ is an operation that sets all non-item entities to $-\infty$, and $\boldsymbol{p}_e \in \mathbb{R}^{|\mathcal{E}|}$.

**Contextual-level Representation.** Entity-level representation and the resulting recommendation only concern what entities have appeared in the context but cannot fully reflect user preferences. For example, if a user says "I do not like A!". We cannot capture such a negative opinion towards "A" through entity-level representation alone. To partly address the problem and to incorporate more semantic- and discourse-level context for recommendation, we further use the context representation $\boldsymbol{H}^C$ computed in §3.2 to yield semantic-aware prediction. Specifically, we average the context representation over the sequence as context-level representation for $C$ and put it through an MLP layer to give the prediction:

$$\boldsymbol{p}_c = \mathrm{softmax}(\mathrm{MLP}(\sum_{j=1}^{|C|} \boldsymbol{h}_j^C)) \tag{7}$$

where $\boldsymbol{h}_j^C$ indicates the representation for the $j$-th token in the context representation $\boldsymbol{H}^C$, $|C|$ is the total length of context $C$, and $\boldsymbol{p}_c \in \mathbb{R}^{|\mathcal{E}|}$.

**Joint Recommendation.** The final recommendation based on the above two components is:

$$\boldsymbol{p}_{rec} = \mu \cdot \boldsymbol{p}_e + (1 - \mu) \cdot \boldsymbol{p}_c \tag{8}$$

where $\mu$ $(0 \leq \mu \leq 1)$ is a hyper-parameter to balance between the two kinds of recommendations.

The learning objective for the recommendation module can be summarized as:

$$\mathcal{L}_{rec} = -\sum_{i=1}^{M} \log p_e(r_i) + \log p_c(r_i) \tag{9}$$

where $M$ is the number of items that need to be recommended and $r_i \in \mathcal{I}$ is the target item in the $i$-th recommendation. $p_e(r_i)$ and $p_c(r_i)$ are the corresponding prediction probabilities of the target item from the entity-level and contextual-level recommendation components, respectively.

### 3.4 Module Integration

We introduce an integration mechanism to incorporate the recommendation module's knowledge and guide the generation module to generate responses that are more consistent with the user's preference. Inspired by Chen et al. (2019), we add a *vocabulary bias* to the top decoder predictions. Different from their work, our vocabulary bias directly comes from the recommendation probabilities $\boldsymbol{p}_{rec}$ in Eq 8:

$$\boldsymbol{b}_u = [\boldsymbol{0}; \mathcal{G}(\boldsymbol{p}_{rec})] \tag{10}$$

where $\mathbf{0}$ is a $\mathcal{V}$-dimensional zero vector, $\mathcal{G}(\cdot)$ is an index selection operation to select items from entities($\mathcal{G} : \mathbb{R}^{|\mathcal{E}|} \rightarrow \mathbb{R}^{|\mathcal{I}|}$), and $[;]$ means concatenation. This makes the bias the same dimension as our generation output (i.e. $|\mathcal{V}'| = |\mathcal{V}| + |\mathcal{I}|$).

We dynamically add the bias $\boldsymbol{b}_u$ during generation based on the top predictions in each time stamp $t$. The effect of this is that for a generation token $y_t$ that is an item in $\mathcal{I}$, we add its recommendation probability $p_{rec}(y_t)$ to the original generation probability $p(y_t)$. In this way, the generation module can generate recommendation-aware responses.

Finally, the total objective for our model is:

$$\mathcal{L} = \mathcal{L}_{gen} + \gamma \mathcal{L}_{rec} \tag{11}$$

where $\gamma$ is a hyper-parameter that balances the two objectives.

## 4 Experimental Setup

**Datasets.** To empirically evaluate the proposed approach, we conduct experiments on two datasets, namely, ReDial (Li et al., 2018) and OpenDialKG (Moon et al., 2019). ReDial is centered around movie recommendation. OpenDialKG consists of conversations that are mainly in four domains: movie, book, sports and music. We split them into training, validation and test at ratios of 80%:10%:10% and 75%:15%:15% by following Li et al. (2018) and Moon et al. (2019). More statistics about the datasets can be found in Appendix.

**Parameter Setting.** We implement our models based on FAIRSEQ framework[1] (Ott et al., 2019), and train on an NVIDIA 3090 GPU. For the RGCN-based recommendation module, we set both the entity embedding size and the hidden representation size to 128. The layer number for R-GCN is 1 and the normalization factor $Z_{e,r}$ is set to 1 following Chen et al. (2019). For the BART-based generation module, we adopt a BART base model consisting of 6 layers of encoder and decoder. The hidden dimension of these encoders and decoders are set to 768. We set the max tokens to 4096 with an update frequency of 4. We adopt Adam optimizer with a 5e-3 learning rate (and 5e-5 learning rate for BART-related modules) and 1000 warm-up updates followed by a polynomial decay scheduler. We adopt diverse beam search (Vijayakumar et al., 2016) mechanism in generation with a beam size of 4 and diverse beam group number of 2. All the hyper-parameters are determined by grid-search based on validation performance. More detailed parameter setting can be found in Appendix.

**Baselines and Comparisons.** For ReDial, we compare several competitive baselines that include:
- **ReDial** (Li et al., 2018) consists of an HRED-based (Sordoni et al., 2015) generation module, and an auto-encoder based recommender module.
- **KBRD** (Chen et al., 2019) uses DBpedia to recommend and adopts a transformer based generation, where knowledge graph (KG) information serves as word bias to assist the generation.
- **KGSF** (Zhou et al., 2020a) uses MIM (Viola and Wells III, 1997) to align the semantic spaces of word- and entity-level KGs. It adopts a transformer encoder and a fused knowledge-enhanced decoder.
- **RevCore** (Lu et al., 2021) performs review-enriched and entity-based recommendation and use a review-attentive encoder-decoder for generation.

We do re-implementations of KBRD and RevCore models, denoted as **EM-SA** and **EM-SA-Rev**, where EM-SA means entity modeling with self-attention[2]. We also test performance of the variants to our model: **BART**, **EM-TA**, **EM-TA-BART**. BART refers to model that uses contextual-based recommendation (described in §3.3) only. EM-TA means using entity-based recommendation with time-aware attention only. EM-TA-BART means the joint model of the above two models.

For the OpenDialKG dataset, we compare the models (seq2seq, Tri-LSTM, Ext-ED, DialKG Walker) that are described in Moon et al. (2019) and skip the introduction to them for saving spaces. Please refer to Moon et al. (2019) for the details.

**Evaluation Metrics.** We evaluate the recommendation and generation separately. For recommendation, we adopt Recall@K scores: K = 1, 10, 50 for ReDial by following Chen et al. (2019), and K = 1, 3, 5, 10, 25 for OpenDialKG by following Moon et al. (2019). Recall@K indicates whether the predicted top-K items contain the ground truth recommendation items. For generation, apart from Dist-n (n=2, 3, 4) and PPL scores reported in (Lu et al., 2021), we also report the case-insensitive BLEU-n (n=2, 4) scores[3]. For a fair comparison, we calculate the PPL scores via a widely used off-the-shelf package KenLM[4], as the PPL scores are very different when using different vocabulary.

[1] https://github.com/pytorch/fairseq

[2] Its attention weights are calculated by a learnable matrix.

[3] We use NLTK package (https://www.nltk.org) to calculate the BLEU scores.

[4] https://kheafield.com/code/kenlm/

| Models | Input | R@1 | R@10 | R@50 |
|---|---|---|---|---|
| Baselines | | | | |
| ReDial | C+Sentiment | 2.4 | 14.0 | 32.0 |
| KBRD | C+TE+EK | 3.1 | 15.0 | 33.6 |
| KGSF | C+TE+EK+WK | 3.9 | 18.3 | 37.8 |
| RevCore | C+TE+EK+Rev | 6.1 | 23.6 | 45.4 |
| Re-implementation | | | | |
| EM-SA(KBRD) | C+TE+EK | 3.3 | 16.3 | 32.6 |
| EM-SA-Rev$^\dagger$(RevCore) | C+TE+EK+Rev | 3.3 | 16.6 | 33.8 |
| Our Models | | | | |
| BART | C | 3.0 | 16.4 | 35.0 |
| EM-TA | C+EK | 4.6 | 18.3 | 34.1 |
| EM-TA-BART | C+EK | 5.5 | 21.2 | 40.0 |
| EM-TA | C+TE+EK | 5.2 | 18.2 | 34.6 |
| EM-TA-BART | C+TE+EK | 5.8 | 20.8 | 40.2 |

Table 1: **Recommendation** results (in %) on ReDial dataset. “C”, “TE”, “EK”, “WK”, “Rev”, “EM-SA” and “EM-TA” are short form for “Context”, “Text Entity”, “Entity-level Knowledge”, “Word-level Knowledge”, “Review”, “Entity Modeling with Self-Attention” and “Entity Modeling with Time-aware Attention”, respectively. Rev$^\dagger$ indicates the extracted items from the retrieved reviews may be not the same as RevCore.

## 5 Experimental Results

In this section, we first report the main comparison results on recommendation and generation in §5.1 and §5.2, respectively. Then we further analyze the effectiveness of our model in §5.3.

### 5.1 Recommendation Result Comparisons

We first present the main comparison results on ReDial in §5.1.1. To further verify the effectiveness of our model in multi-domain dataset, we conduct experiments on OpenDialKG and report the comparison results in §5.1.2.

#### 5.1.1 Results on ReDial

Table 1 shows the recommendation results of our models, the baselines and our re-implementation of some baselines. We can draw the following observations from the results:

- *Adding more external knowledge can improve the recommendation performance.* We can see that among the baselines, KBRD adds the entity-level knowledge (EK), while KGSF and RevCore further incorporate word-level knowledge (WK) and item reviews (Rev). All the external knowledge introduces considerable improvement, demonstrating the efficiencies of the external knowledge.

- *More external knowledge like the item reviews introduces greater difficulty in reproduction and less generalization.* We have tried to re-implement the results of KBRD and RevCore baselines (the re-implementation results are also shown in Table 1). We find that we can easily re-implement similar results of KBRD but cannot achieve improvement when further incorporating item review information following RevCore. The reasons can be twofold. First, incorporating multiple external knowledge introduces more challenges to balance them. The other is that the method introduced in RevCore requires much extra effort to collect and annotate the reviews (some items may even lack reviews) and train a sentiment-aware retrieval model (Lu et al., 2021), which makes it difficult to reproduce similar results and become less generalizable to other domains.

- *Time-aware attention can better summarize user preference than self-attention.* Our models with time-aware attention perform significantly better than the models with self-attention. For example, EM-TA achieves 1.9% higher Recall@1 compared to EM-SA in the same input (C+TE+EK) situation. This validates our intuition that the recently appeared items are more important for reflecting user preference, as well as the effectiveness of our designed time-aware attention.

- *BART-based representations are helpful.* We are the first to finetune a pretrained BART model and utilize the representations for recommendation. As we can see in Table 1, the simplest BART model achieves 35.0% Recall@50 while the KBRD model that incorporates external knowledge graph gets 33.6 Recall@50. We can also find that our models with time-aware attention show good improvements in all metrics after being enhanced with BART representations. Both indicate that contextual-level representations extracted by BART can reflect user preference that entity-level representations cannot capture. Fig. 2(a) shows more detailed analysis.

- *Text entities are effective in capturing most relative items.* Our models with additional text entities (TE) as input can achieve better Recall@1 compared with the same models without TE, while keeping similar Recall@10 and Recall@50. This means that text entities help re-rank the top predictions and find the most relative items.

#### 5.1.2 Results on OpenDialKG

Apart from ReDial that focuses on movie recommendation, we also examine our recommendation performance in a multi-domain dataset, OpenDialKG, to show the generalizability of our model. The results are displayed in Table 2. Our model

| Models | R@1 | R@3 | R@5 | R@10 | R@25 |
|---|---|---|---|---|---|
| Baselines | | | | | |
| seq2seq | 3.1 | 18.3 | 29.7 | 44.1 | 60.2 |
| Tri-LSTM | 3.2 | 14.2 | 22.6 | 36.3 | 56.2 |
| Ext-ED | 1.9 | 5.8 | 9.0 | 13.3 | 19.0 |
| DialKG Walker | 13.2 | 26.1 | 35.3 | 47.9 | 62.2 |
| Our Models | | | | | |
| BART | 5.8 | 19.7 | 31.0 | 45.5 | 57.8 |
| EM-SA | 10.9 | 21.2 | 30.3 | 41.6 | 53.2 |
| EM-TA | 16.0 | 28.9 | 34.3 | 45.1 | 57.9 |
| EM-TA-BART | **18.0** | **33.5** | **41.5** | **50.0** | **64.8** |

Table 2: **Recommendation** results (in %) on OpenDialKG. 'EM-SA" and "EM-TA" are short form for "Entity Modeling with Self-Attention" and "Entity Modeling with Time-aware Attention", respectively.

with time-aware attention and BART-enhanced representations achieves the best performance compared to all the other methods. We can observe similar trends among the different variants as those in ReDial, e.g., time-aware attention is better than self-attention and BART representations help improve all of the metrics. This validates that our method generalizes well to other domains.

### 5.2 Generation Result Comparisons

**Automatic Evaluation.** We show the generation comparison results in Table 3. To investigate the performance in different scenarios, we display the results of our model with conventional beam search (BS) and diverse beam search (DBS), respectively, together with their results without BART pre-training. We summarize our observations in the following:

• *Our model is able to generate more diverse and fluent responses than the baselines.* As can be seen, our model with beam search achieves the best BLEU and perplexity scores, and our model with diverse beam search yields the highest Dist-n while maintaining comparable BLEU and perplexity.

• *It is challenging to balance diversity and fluency.* The baselines perform differently in terms of Dist-n and perplexity, e.g., KGSF achieves higher Dist-n than KBRD, but its perplexity is worse. We presume that higher diversity requires the models to extract more different patterns to express the content, but organizing them into a fluent response may be challenging. Another example is our model without BART pre-training achieves poor diversity, as it may overfit on the training corpus and tend to generate simple responses. But this also results in its lowest perplexity. Our model with diverse beam search achieves consistently better Dist-n, and maintains relatively lower perplexity, demonstrating the superiority of our model.

| Models | Dist-2 | Dist-3 | Dist-4 | BLEU2 | BLEU4 | PPL |
|---|---|---|---|---|---|---|
| Transformer | 14.8 | 15.1 | 13.7 | - | - | - |
| ReDial | 22.5 | 23.6 | 22.8 | 17.8 | 7.4 | 61.7 |
| KBRD | 26.3 | 36.8 | 42.3 | 18.5 | 7.4 | 58.8 |
| KGSF | 28.9 | 43.4 | 51.9 | 16.4 | 7.4 | 131.1 |
| RevCore | 42.4 | 55.8 | 61.2 | - | - | - |
| Ours+BS | 35.8 | 49.9 | 57.7 | **19.1** | **9.3** | 52.1 |
| - BART PT | 8.5 | 10.9 | 12.3 | 18.6 | 8.7 | **30.7** |
| Ours+DBS | **45.7** | **65.3** | **76.1** | **19.1** | 8.9 | 54.8 |
| - BART PT | 13.9 | 19.8 | 23.8 | 18.6 | 8.2 | 43.9 |

Table 3: **Generation** results (in %) on the ReDial dataset. "BS" refers to beam search, "DBS" refers to diverse beam search, and "PT" refers to pre-training.

| Models | Fluency | Informativeness | Coherence |
|---|---|---|---|
| HUMAN | **1.95** | **1.71** | **1.71** |
| ReDial | 1.92 | 1.32 | 1.23 |
| KBRD | **1.95** | 1.39 | 1.31 |
| KGSF | 1.91 | 1.02 | 0.95 |
| Ours+BS | **1.95** | 1.54 | 1.66 |
| Ours+DBS | 1.92 | 1.64 | 1.64 |

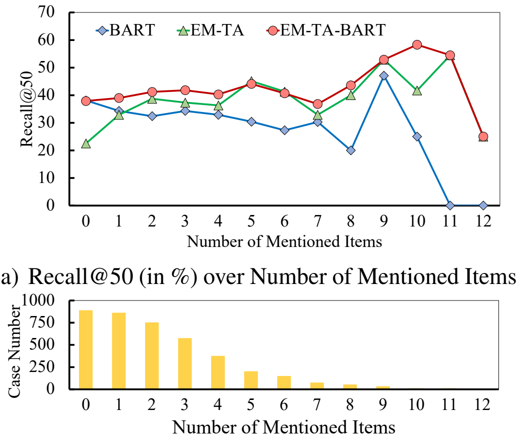
Table 4: **Human evaluation** of the **generation** results on the ReDial dataset. "BS" refers to beam search, "DBS" refers to diverse beam search. All the metrics are in the scale of [0, 2]. The overall Cohen's kappa coefficient is larger than 0.65.

• *Dist-n scores highly depend on the searching strategy.* Our model performs much better in terms of Dist-n when applying diverse beam search compared with using conventional beam search. What is more, different configuration (e.g., length penalty) applied during generation also affects the scores. More analysis can be found in Appendix.

**Human Evaluation.** We adopt a human evaluation to examine the generation results from a different perspective and display the results in Table 4. The evaluation details are given in the Appendix. From Table 4, all the models generate responses with high fluency, but perform differently regarding informativeness and coherence. The baselines are more likely to produce safe responses (short and repetitive) while our model can generate more informative and coherent responses.

### 5.3 Further Analysis

**Effectiveness of Contextual-level Representation.** To investigate how contextual-level representations influence recommendation, we show the Recall@50 scores with varying mentioned items numbers in context in Fig. 2(a). We can observe that the BART model performs better than EM-TA when the mentioned number is 0 or 1. Such phe-



(a) Recall@50 (in %) over Number of Mentioned Items

(b) Number of cases over Number of Mentioned Items

Figure 2: Recall@50 of the models and number of cases over the number of mentioned items in context.

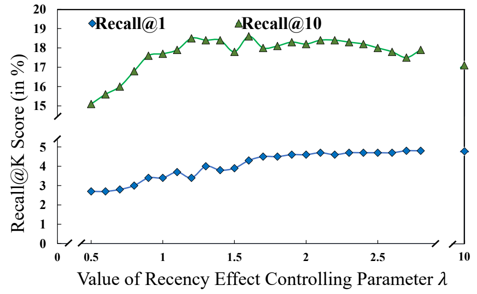


Figure 3: Change of Recall scores when using different values of $\lambda$, which controls the recency effect.

nomenon is desired, since when the item history information is rare or even missing, entity-level representations are not sufficient to produce reliable recommendations while contextual-level representations can capture useful information from the text. Then we find EM-TA performs consistently better than BART when the mentioned number increases. Because the increasing mentioned item number also means the context becomes longer, and the model might be not able to handle the long context, especially when the mentioned item number is larger than 10. This means that contextual-level representations are useful in the cold-start scenario (not a rare situation as shown in Fig. 2(b)), which is a shortcoming for entity-level recommendation. Therefore, combining the both representations (i.e., EM-TA-BART model) yields the best performance.

**Effectiveness of Time-Aware Attention.** We present the Recall@1 and Recall@10 (Recall@50 shows similar trend with Recall@10) scores with varying values of $\lambda$, which can control the effect of recency introduced in Eq. (5) of Fig. 3. $\lambda < 1$ indicates the earlier appeared items are more important, which results in quick performance drops. And the performance increases when $\lambda > 1$ compared to $\lambda = 1$. This validates the intuition that more recently appeared items contribute more to the recommendation. On the other hand, Recall@1 and Recall@10 present different trends when $\lambda$ increases – Recall@1 consistently increases while Recall@10 begins to drop when $\lambda > 2$. This is because when $\lambda$ is too large, time-aware attention tends to concern with the most recently appeared item. This is helpful in finding the most relative items but hurts the overall recommendation.

| Models | Inputs | PT | R@1 | R@10 | R@50 |
|---|---|---|---|---|---|
| BART | C | ✓ | 3.0 | 16.4 | 35.0 |
| | | ✗ | 2.6 | 14.5 | 31.7 |
| EM-TA-BART | C+EK | ✓ | 5.5 | 21.2 | 40.0 |
| | | ✗ | 5.2 | 21.0 | 39.4 |
| EM-TA-BART | C+TE+EK | ✓ | 5.8 | 20.8 | 40.2 |
| | | ✗ | 5.4 | 20.4 | 39.0 |

Table 5: Recommendation results (in %) of our models when using BART pre-training (PT) or not.

**Effects of BART Pre-training.** We examine the effects of BART pre-training (PT) for recommendation and generation. For recommendation, we list the recall scores of our different model variants with and without BART PT in Table 5. As can be seen, all the models perform worse when removing PT, as PT on large-scale monolingual datasets helps the models learn better semantic features. When joint recommending with entity-level representations (i.e., EM-TA-BART model), the performance degradation becomes less, demonstrating the two kinds of information can complement each other.

For generation, we have listed the ablation without PT in Table 3. As we can see, models without PT show poor performance on Dist-n metric as models tend to be overfitting and cannot generate diverse responses based on a small dataset.

## 6 Conclusion

In this work, we propose to capture both entity-level and contextual-level representations to improve the conversational recommender system, where a time-aware attention is designed to emphasize the recently appeared items and a pre-trained BART is used to enhance context modeling. Experiments show that the proposed model can achieve comparable performance with less external knowledge and generalizes well to other domains. Further analyses also examine the effectiveness of the model in different scenarios.

## References

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd international acm sigir conference on research and development in information retrieval*, pages 475–484.

Qibin Chen, Junyang Lin, Yichang Zhang, Ming Ding, Yukuo Cen, Hongxia Yang, and Jie Tang. 2019. Towards knowledge-based recommender dialog system. 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)*, pages 1803–1813, Hong Kong, China. Association for Computational Linguistics.

Konstantina Christakopoulou, Alex Beutel, Rui Li, Sagar Jain, and Ed H Chi. 2018. Q&r: A two-stage approach toward interactive recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 139–148.

Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards conversational recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 815–824.

Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N Mendes. 2013. Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems*, pages 121–124.

Yang Deng, Yaliang Li, Fei Sun, Bolin Ding, and Wai Lam. 2021. Unified conversational recommendation policy learning via graph-based reinforcement learning. *arXiv preprint arXiv:2105.09710*.

Shirley Anugrah Hayati, Dongyeop Kang, Qingxiaoyang Zhu, Weiyan Shi, and Zhou Yu. 2020. Inspired: Toward sociable recommendation dialog systems. *arXiv preprint arXiv:2009.14306*.

Dongyeop Kang, Anusha Balakrishnan, Pararth Shah, Paul Crook, Y-Lan Boureau, and Jason Weston. 2019. Recommendation as a communication game: Self-supervised bot-play for goal-oriented dialogue. *arXiv preprint arXiv:1909.03922*.

Wenqiang Lei, Xiangnan He, Yisong Miao, Qingyun Wu, Richang Hong, Min-Yen Kan, and Tat-Seng Chua. 2020a. Estimation-action-reflection: Towards deep interaction between conversational and recommender systems. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 304–312.

Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020b. Interactive path reasoning on graph for conversational recommendation. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 2073–2083. ACM.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. 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*, pages 7871–7880, Online. Association for Computational Linguistics.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670.

Raymond Li, Samira Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards deep conversational recommendations. *arXiv preprint arXiv:1812.07617*.

Shijun Li, Wenqiang Lei, Qingyun Wu, Xiangnan He, Peng Jiang, and Tat-Seng Chua. 2020. Seamlessly unifying attributes and items: Conversational recommendation for cold-start users. *arXiv preprint arXiv:2005.12979*.

Shuai Li, Alexandros Karatzoglou, and Claudio Gentile. 2016. Collaborative filtering bandits. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 539–548.

Zeming Liu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, and Ting Liu. 2020. Towards conversational recommendation over multi-type dialogs. *arXiv preprint arXiv:2005.03954*.

Yu Lu, Junwei Bao, Yan Song, Zichen Ma, Shuguang Cui, Youzheng Wu, and Xiaodong He. 2021. Revcore: Review-augmented conversational recommendation. *arXiv preprint arXiv:2106.00957*.

Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. Opendialkg: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 845–854.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Xuhui Ren, Hongzhi Yin, Tong Chen, Hao Wang, Zi Huang, and Kai Zheng. 2021. Learning to ask appropriate questions in conversational recommendation. *arXiv preprint arXiv:2105.04774*.

Xuhui Ren, Hongzhi Yin, Tong Chen, Hao Wang, Nguyen Quoc Viet Hung, Zi Huang, and Xiangliang Zhang. 2020. Crsal: Conversational recommender systems with adversarial learning. *ACM Transactions on Information Systems (TOIS)*, 38(4):1–40.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer.

Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *proceedings of the 24th ACM international on conference on information and knowledge management*, pages 553–562.

Yueming Sun and Yi Zhang. 2018. Conversational recommender system. In *The 41st international acm sigir conference on research & development in information retrieval*, pages 235–244.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*.

Paul Viola and William M Wells III. 1997. Alignment by maximization of mutual information. *International journal of computer vision*, 24(2):137–154.

Hu Xu, Seungwhan Moon, Honglei Liu, Bing Liu, Pararth Shah, Bing Liu, and Philip Yu. 2020. User memory reasoning for conversational recommendation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5288–5308, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Kerui Xu, Jingxuan Yang, Jun Xu, Sheng Gao, Jun Guo, and Ji-Rong Wen. 2021. Adapting user preference to online feedback in multi-round conversational recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 364–372.

Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W Bruce Croft. 2018. Towards conversational search and recommendation: System ask, user respond. In *Proceedings of the 27th acm international conference on information and knowledge management*, pages 177–186.

Kun Zhou, Wayne Xin Zhao, Shuqing Bian, Yuanhang Zhou, Ji-Rong Wen, and Jingsong Yu. 2020a. Improving conversational recommender systems via knowledge graph based semantic fusion. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1006–1014.

Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, Xiaoke Wang, and Ji-Rong Wen. 2020b. Towards topic-guided conversational recommender system. *arXiv preprint arXiv:2010.04125*.

# Appendix

## A Datasets and Parameter Setting

| | ReDial | OpenDialKG |
|---|---|---|
| Number of conversations | 10,006 | 13,802 |
| Number of utterances | 182,150 | 91,209 |
| Knowledge Graph | DBpedia | DBpedia |
| Domains | Movie | Movie, Book |

Table 6: Statistics of ReDial and OpenDialKG datasets.

We first show the basic statistics of the two datasets in Table 6. Then we show the detailed parameter search space and best assignment in Table 7. The parameter number of our model is 269M. The training time of one epoch is around 22 minutes when the model is trained an NVIDIA 3090 GPU with a max token number of 4096 and an update frequency of 4. The model needs around 5 epochs to achieve the best performance on the validation set.

## B Human Evaluation Details

We randomly sampled 100 context-response pairs from the test set and collected the corresponding generation results of our models as well as the baseline models. We then employ two crowd-workers to score the results on the scale of [0, 1, 2], where higher scores indicate better quality. Following prior studies, we also evaluate three aspects:

- **Fluency**: whether a response is in a proper English grammar and easy to understand.
- **Informativeness**: whether a response contains meaningful information. The "safe responses" are treated as uninformative as they may be repetitive and meaningless.
- **Coherence**: whether a response is coherent with the context, i.e., the discussion content should be consistent.

The scoring details are shown in Table 8 following one of the previous work.

## C More Analysis

**Limitation of Dist-n Metrics.** As we find that search strategies seriously affect the Dist-n metrics, we present more analysis on them by setting different values of length penalty (a hyper-parameter that can control the lengths of final generated results) when generating the responses. We display the results of Dist-2 and BLEU2 for EM-TA-BART model with beam search setting in Fig. 4 (other metrics are in similar trends). We can find that the generated lengths also affect much on the Dist-n, since longer responses allow more different tokens to be generated. However, this is not expected as not the longer the better. Therefore, other metrics including human evaluation are desired to explicitly evaluate generation performance.

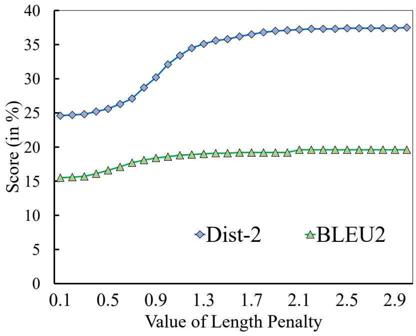


Figure 4: Change of Dist-2 and BLEU2 scores when using different length penalty. Larger length penalty ($> 1$) indicates allowing generating longer responses.

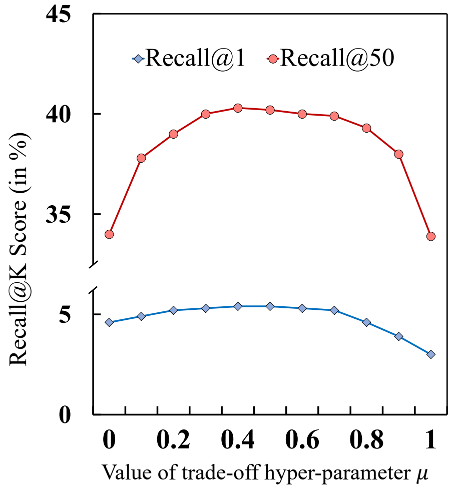


Figure 5: Change of Recall@1 and Recall@50 scores over different values of trade-off parameter $\mu$.

**Trade-off between Entity-level and Contextual-level Representations.** We examine the effects of the hyper-parameter $\mu$ in Eq. (8) by setting its value from 0 (only entity-level representations) to 1 (only contextual-level representations) and display the results of EM-TA-BART model with C+EK input in Fig. 5. As can be seen, Recall@50 is significantly improved when $\mu$ changes from 0 to 0.1 (or 1 to 0.9). This validates that the two representations capture user preferences from a different perspective and can complement each other. The best result is achieved with $\mu = 0.5$, showing that both representations are important.

| Hyper-parameter | Search Space | Best Assignment |
|---|---|---|
| RGCN entity embedding size | {128, 256} | 128 |
| RGCN hidden representation size | {128, 256} | 128 |
| RGCN layer number | 1 | 1 |
| Normalization factor $Z_{e,r}$ | 1 | 1 |
| BART layer number | 6 | 6 |
| BART hidden dim | 768 | 768 |
| Max token number | {2048, 4096, 8192, 10240} | 4096 |
| Update frequency | {1, 2, 4, 8} | 4 |
| LR for recommendation | [1e-4, 2e-4, ..., 8e-3] | 5e-3 |
| LR for generation | [1e-5, 2e-5, ..., 5e-4] | 5e-5 |
| Warm-up updates | {200, 400, 600, 800, 1000, 2000} | 1000 |
| Patience | 5 | 5 |
| $\mu$ trade off | [0, 0.1, 0.2, ..., 1.0] | 0.5 |
| beam size | {2, 4, 6, 8} | 4 |
| diverse beam group number | {2, 4} | 2 |
| length penalty | [0.1, 0.2, ..., 3] | 1.5 |

Table 7: Hyper-parameter Search Space and Best Assignment.

| Score | Fluency |
|---|---|
| **0** | The response has many grammar mistakes.<br>The response is hard to understand. |
| **1** | The response has minor grammar mistakes.<br>Some part of the response is hard to understand. |
| **2** | The response is in correct grammar and easy to understand. |

| Score | Coherence |
|---|---|
| **0** | The response is not related with the context.<br>The response simply repeats the context.<br>The response has obvious conflicts with the context. |
| **1** | The response has minor conflicts with the context.<br>There are some minor logic conflicts in the response. |
| **2** | The response is coherent with the context. |

| Score | Informativeness |
|---|---|
| **0** | The response does not contain any information.<br>This response just repeats the context and fails to bring any additional information.<br>The information is invalid, as the coherence score is 0. |
| **1** | The information has conflicts with common sense.<br>There are factual errors in the response. |
| **2** | The response has appropriate and correct information. |

Table 8: Scoring details for human evaluation.