HOOPS : Human-in-the-Loop Graph Reasoning for Conversational Recommendation

Zuohui Fu[†*], Yikun Xian[†*], Yaxin Zhu[†], Shuyuan Xu[†], Zelong Li[†]
Gerard de Melo[‡], Yongfeng Zhang[†]
[†]Rutgers University, New Brunswick, NJ [‡]HPI/Univ. of Potsdam, Germany
zuohui.fu@rutgers.edu,siriusxyk@gmail.com,{yz956,shuyuan.xu,zelong.li}@rutgers.edu
gdm@demelo.org,yongfeng.zhang@rutgers.edu

ABSTRACT

There is increasing recognition of the need for human-centered AI that learns from human feedback. However, most current AI systems focus more on the model design, but less on human participation as part of the pipeline. In this work, we propose a Human-in-the-Loop (HitL) graph reasoning paradigm and develop a corresponding dataset named HOOPS for the task of KG-driven conversational recommendation. Specifically, we first construct a KG interpreting diverse user behaviors and identify pertinent attribute entities for each user–item pair. Then we simulate the conversational turns reflecting the human decision making process of choosing suitable items tracing the KG structures transparently. We also provide a benchmark method with reported performance on the dataset to ascertain the feasibility of HitL graph reasoning for recommendation using our developed dataset, and show that it provides novel opportunities for the research community.

KEYWORDS

Human-in-the-Loop Learning; Graph Reasoning; Recommender Systems; Conversational Recommendation; Explainable Recommendation

ACM Reference Format:
Zuohui Fu, Yikun Xian, Yaxin Zhu, Shuyuan Xu, Zelong Li, Gerard de Melo, Yongfeng Zhang. 2021. HOOPS : Human-in-the-Loop Graph Reasoning for Conversational Recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3404835.3463247

1 INTRODUCTION

Given the increasing recognition of human-centered AI as a new paradigm for AI, Human-in-the-Loop (HitL) learning has emerged as an essential way of leveraging the power of both machine intelligence and human intelligence to enable collaborative human–machine-driven reasoning and decision making [6, 12, 30]. An intuitive example is the *Guess the Number* game [13, 14]. In this game, the user has a secrete number in mind. The system interacts with the user through conversation by asking questions in multiple rounds to narrow down the range of possible numbers until the correct answer is identified, while the user responds to the questions by telling the system whether the current guess is too low or too high. In this process, the human user serves as the supervision to the system. The benefit of such human-involved feedback is that it substantially reduces the search space in the guessing process and improves the efficiency of algorithms.

In real-world human-centered tasks such as conversational recommendation, one can analogously consider a graph reasoning problem that involves *guessing* what might be the ideal item in a user's mind by asking the user questions and searching over a heterogeneous user-item-attribute graph [31, 45]. As illustrated in Fig. 1, starting from a user node, the system, at every step, needs

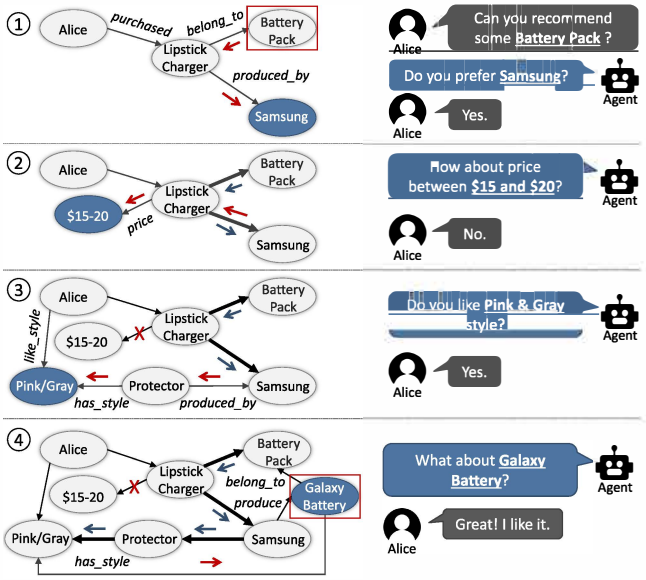


Figure 1: We regard conversational recommendation as a typical concretization of the HitL graph reasoning paradigm, aiming to predict the next suitable question and make recommendations in multi-round dialogue. The user feedback allows pruning off irrelevant candidates.

*Both authors contributed equally to this work.

SIGIR '21, July 11–15, 2021, Virtual Event, Canada

ACM ISBN 978-1-4503-8037-9/21/07...$15.00
https://doi.org/10.1145/3404835.3463247

to determine how to move to a promising neighboring node and finally arrives at a potential item node of interest. For traditional graph reasoning problems, the search space is often prohibitively large [38, 49], which makes it rather challenging and inefficient to *guess* a correct answer (i.e., arrive at a correct target node in the graph). Therefore, in this work, we propose the novel HitL graph reasoning paradigm, where the human is allowed to provide feedback to help the system prune irrelevant actions and quickly locate a correct path towards a target node.

The HitL graph reasoning paradigm possesses the following properties. First, unlike the traditional decision making process that heavily relies on past interaction history, HitL should be **hybrid** by integrating static features as well as real-time human intervention as essential forms of inputs. Second, in practice, humans often proceed in a **coarse-to-fine** manner to gradually make their decisions. For example, people answer questions by first skimming the text, identifying key ideas, and then carefully reading specific parts to obtain an answer [22, 39]. Similarly, HitL graph reasoning will first pursue attribute nodes describing broader concepts. Subsequently, with more interaction loops with the user, the system will gradually gain a better understanding of the specific user requirements and preferences pertaining to the relevant goal entity to be chosen. Third, path reasoning through the HitL paradigm is expected to highlight the **transparency** of the decision making process in the sense that the system exposes its reasoning process with user feedback by revealing the corresponding paths in the graph [38]. This sort of transparency enables users to witness not only what the systems provides in response to their input but also how it updates its reasoning and whether their input is incorporated as they expect.

In this work, we consider the task of product recommendation as a concrete example to demonstrate how intelligent systems can benefit from our HitL graph reasoning framework, and we also provide a new benchmark dataset to study the problem. In order to incorporate the human participation in the system, we consider a multi-round conversational recommender system (CRS) as a typical implementation. Our novel dataset integrates product information as well as diverse user participation and historical records. At the same time, our dataset follows natural coarse-to-fine conceptual resolution to gradually infer the user interests starting from broader interests, e.g., categories or brands. Through multiple rounds of interaction, the system gradually gains a more detailed understanding of specific user requirements and preferences pertaining to the relevant products to be chosen. Last but not least, in order to make the user–agent interaction loop more transparent, we draw on a unified knowledge graph based on the Amazon review corpus [26] such that the conversational system can better assist users to retrieve the best-suited products through an explicit graph reasoning process. To show the applicability of the datasets, we also provide a baseline method with reported performance over three tasks. The contributions of this paper are threefold. 1) We propose a novel human-in-the-loop (HitL) graph reasoning paradigm with three important properties. 2) We construct a new dataset for conversational recommendation under the proposed framework. 3) We provide a new method and its performance on the dataset, which can be used for future research on human-in-the-loop learning.

	Cellphones	Grocery	Toys & Games	Automotive
#Entities	278,198	271,855	437,897	444,545
#Relations	45	45	71	73
#Triples	3,724,724	4,452,234	6,705,842	5,703,094
#Interactions	607,673	709,280	1,178,943	1,122,776
#Utterances	2,043,988	2,424,103	3,339,771	3,830,556

Table 1: Statistics of our dataset on four domains.

2 HITL GRAPH REASONING PARADIGM

A unified knowledge graph $\mathcal{G} = \{(e, r, e') \mid e, e' \in \mathcal{E}, r \in \mathcal{R}\}$ is defined to be a set of triples with an entity set $\mathcal{E}$ and relation set $\mathcal{R}$. The entity set consists of three types of nodes, source nodes ($\mathcal{U} \subseteq \mathcal{E}$), target nodes ($\mathcal{V} \subseteq \mathcal{E}$) and descriptive nodes ($\mathcal{A} = \mathcal{E} \setminus \mathcal{U} \cup \mathcal{V}$). A path L over the graph is a sequence of entities and relations, i.e., $L = \{e_0, r_1, e_1, \ldots, e_{|L|-1}, r_{|L|}, e_{|L|}\}$. For traditional graph reasoning tasks, given a source node $u \in \mathcal{U}$, the goal is to find a multi-step path L whose end node $e_{|L|} \in \mathcal{V}$ is regarded as the prediction. To facilitate human interaction in the graph reasoning, at each step t, the agent generates a question Q_t based on the traversed path to solicit help from the user. The user provides a response R_t that may be a direct answer to the question but may also consist of ambiguous statements or other arbitrary dialogue discourse. Given a vocabulary V, we define $Q_t, R_t \in V^{d_w}$ with d_w as the maximum length of a question or response. Given a source node $u \in \mathcal{U}$ and an unknown target node $v \in \mathcal{V}$, the workflow of the HitL graph reasoning paradigm is defined as follows. At every step $t+1$, given the traversed path $L_t = \{u, r_1, e_1, \ldots, r_{|L_t|}, e_{|L_t|}\}$ ($|L_t| \geq t$) and past human–agent interactions $Q_0, Q_1, R_1, \ldots, Q_t, R_t$, the agent aims to (i) find a k-hop path $L^{(t+1)}$ from $e_{|L_t|}$ to a descriptive node in $\mathcal{A}$, (ii) ask the user a question Q_{t+1} conditioned on $L^{(t+1)}$ and receive a response R_{t+1}, and (iii) make a decision by predicting top K target nodes $\{v_t^{(1)}, \ldots v_t^{(K)}\} \subseteq \mathcal{V}$. By the end of turn $t+1$, based on the user response, the agent can form the new reasoning path by either extending the path with $L_{t+1} = L_t \cup L^{(t+1)}$ (i.e., move to the next descriptive node) or keeping the old one $L_{t+1} = L_t$ (i.e., stay at node $e_{|L_t|}$). Note that when $k = 1$, the agent simply finds the neighboring nodes of $e_{|L_t|}$. The interaction will terminate if the user refuses to continue or the maximum step T is reached.

Conversational Recommendation. The HitL graph reasoning paradigm can be instantiated as follows in this scenario. We consider a source node from the set of users $\mathcal{U}$, a target node from the set of items $\mathcal{I}$, and descriptive nodes $\mathcal{A}$ as attribute entities that either denote properties of items or descriptive words that a user mentions in the conversational turns. When users start a conversation with the agent, thereby initializing the HitL graph reasoning, it is reasonable to expect that they typically begin with broader requirements, such as the preferred category and brand within the descriptive nodes. As illustrated in Figure 1, the agent asks questions based on both **hybrid** user behaviors that integrate past user activity and current user feedback in conjunction with item attribute knowledge, aiming at more engaging and informative entities as the conversation progresses. The entities in $\mathcal{A}$ are transformed into human-readable questions to identify the user needs. Thus, the HitL graph reasoning is expected to find a path that leads to the next potential descriptive node aiming to follow natural **coarse-to-fine** conceptual resolution to gradually narrow

down the user interests. On the one hand, the ultimate goal of HitL graph reasoning is to reach a target node in $\mathcal{V}$, which also corresponds to recommending an item to the user. On the other hand, the system needs to select suitable questions to ask in each round and traverse the graph. This explicit graph traversal also ensures that the reasoning is **transparent**.

3 DATASET CONSTRUCTION

Data Source. To construct a dataset that facilitates this new HitL graph reasoning paradigm, we draw on a recent compilation of Amazon review data [26] that includes extensive user reviews and rich item information. It is subdivided into several categories, each of which covers a separate sub-domain of items from the e-commerce platform and hence can be regarded as an independent benchmark for the task. We pick four categories to construct the datasets, encompassing Cellphones & Accessories, Grocery & Gourmet, Toys & Games, and Automotive (see Table 1).

Graph Construction. To enable graph reasoning, we first construct a knowledge graph (KG) with rich meta-information of user behavior and item meta-data. First, we extract the keywords from user reviews following Zhang et al. [45] and identify appreciated aspects of items from a user's historical records on Amazon. This yields multiple categories of user records (purchases, comments, etc.). We consider the following abundant information as item meta-data: item category, brand, listed features, predefined styles, etc. These two parts constitute the descriptive nodes in the KG. Unlike previous works that simply tie existing recommendation datasets (e.g., MovieLens, LFN-1b, etc.) to a knowledge base (e.g., Freebase [47]) to enrich the item information [34, 35], our constructed KG not only captures copious amounts of item meta-information but also incorporates abundant user interactions with items to support HitL graph reasoning for recommendation. We leverage explicit semantics from user interactions extracted as structured information and KG relations between source nodes, descriptive nodes, and target nodes. Hence, the constructed KG with source nodes as user entities and target nodes as item entities can provide more relevant and supportive information for systems to ask suitable questions regarding the attributes of potential items and drive the transparent graph reasoning paradigm for recommendation.

Coarse-to-Fine Extraction. Instead of providing a *correct* path for graph reasoning, we generate a sequence of ground-truth attribute nodes that describe target item properties. The underlying intuition is that since the conversational system aims to help users gradually figure out their preferences, we assume the system starts from the descriptive nodes with larger degrees, as these are more prominent, well-known, and often more generic. As the conversation loop proceeds, the latent needs of users are progressively clarified such that it becomes easier to consider the descriptive nodes with a smaller degree, i.e., more particular fine-grained ones. In graph theory [25], node degree centrality is among the most prominent measures of node importance over the graph structure. Therefore, we first extract the descriptive entities that are reachable from the given user and item within one or two hops as attribute entities, and then sort them according to the node degrees. The intuition behind this is that a larger degree indicates that the entity carries broader information [27] and is easier for the model to predict, while a smaller degree implies the entity is more specific to a user or item but is harder to predict. The sorted sequence of attribute entities serves as a skeleton for the corresponding dialogue, guiding a coarse-to-fine selection process in which the entities determine which feature is considered in each conversational turn.

Conversation Generation. Instead of directly extracting utterances from user reviews [45], we employ the template approach of Wiseman et al. [36] based on a large data-driven dialogue corpus [5, 10]. We compose the corresponding conversations based on the skeleton formed by the respective sequence of attribute entities, transforming each attribute entities into questions via human-specified English language templates generated from Wiseman et al. [36]. We then randomly determine the user response to the question with clarified answer "Yes/No" or unclear answer "I'm not sure/I don't know" etc. with some predefined probability to mimic real conversations, especially for the sake of modeling users in practical HitL scenarios, where typically they are unclear about their preferences with regard to potential items. It also makes sense to assume that those users seeking assistance rather than directly selecting an item tend to be unfamiliar with the product details and are unable to provide detailed requirements. Therefore, we envision this benchmark as serving as an initial milestone for a practical HitL graph reasoning for recommender systems to tackle before moving on to even more challenging real-life dialogue with disfluencies, ambiguity, inconsistent preferences, and backtracking.

Dataset Construction. The workflow of constructing our dataset is as follows. For each user $u \in \mathcal{U}$ and an item $v \in \mathcal{V}$ purchased by the user, we take as input a sequence of $T+1$ attribute entities $\{e_0, \ldots, e_T\}$, as obtained in the previous step, along with a sequence of corresponding responses $\{R_1, \ldots, R_T\}$. Here, e_0 is the attribute entity identified from the user's initial query in the conversational loop. We first construct the T-turn conversation: $\{Q_0, (Q_1, R_1, e_1), \ldots, (Q_T, R_T, e_T)\}$, where each question Q_t is generated via a predefined template and associated with corresponding entity $e_t \in \mathcal{E}$ for $t = 0, 1, \ldots, T$. Then, we build three candidate sets via negative sampling. For the item candidate set, we randomly sample a subset of N_V items that the user has not purchased. To construct the attribute candidate set at the $T+1$-th turn, we first sample a set of paths from the user u to item v and randomly retrieve N_A nodes from these paths, denoted by $e_1^-, \ldots, e_{N_A}^-$. Thus, the candidate set can be formed as $\{e_{T+1}, e_1^-, \ldots, e_{N_A}^-\}$, where e_{T+1} is the ground-truth attribute entity previously obtained. The corresponding question candidate set is then generated via the templates and the set of attribute candidates. Since we know the ground-truth of the next question $Q_{T+1} = Q(e_{T+1})$, the next entity e_{T+1}, and the purchased item v, binary labels can also be provided indicating whether or not a model makes a correct prediction.

Human Validation. In order to validate whether the constructed conversations follow the coarse-to-fine property, as shown in Figure 2, we illustrate with box plots the degree of entities in coarse-to-fine extracted entities associated with the N-th turn. Specifically, we sampled 40 sub-dialogues from the dataset and shuffled their original order. 20 human raters were asked to rank the questions according to their preference of correct question orders. The observed trends justify our dataset construction. As the dialogue turns increase, the degree of the entity within the turn decreases. Therefore, it is natural to start with broader, more general questions, and

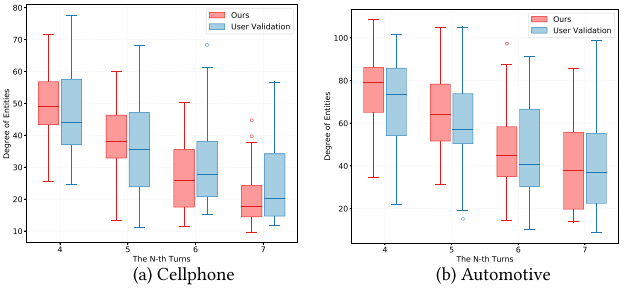
as the conversational loop progresses in a coarse-to-fine manner, the agent will propose increasingly finer-grained questions.

4 BENCHMARK MODEL

Along with the data, we propose a benchmark model consisting of a set of encoders to represent inputs and candidate outputs as low-dimensional vectors and three predictors for different sub-tasks. To learn to conduct the three sub-tasks, each predictor takes multiple encoded vectors as input to reflect the fact that any single sub-task may depend on multiple aspects and that different sub-tasks may mutually benefit each other via joint optimization.

Graph Representation. We represent the historical path by a sequence of attribute entities $x_e = \{e_0, e_1, \ldots, e_T\}$, which is expected to be encoded into a d-dimensional vector $\hat{\mathbf{x}}_e$. Since both descriptive attributes of items and historic behavior of users are captured in the graph $\mathcal{G}$, we first train a TransE model [4] over $\mathcal{G}$, so that each entity in $\mathcal{E}$ and each relation in $\mathcal{R}$ is embedded into a continuous space of dimensionality d_G. Note that graph embedding is not the focus of this paper and any off-the-shelf techniques can be applied here. Therefore, we can represent x_e as a sequence of d_G dimensional vectors, i.e., $\mathbf{x}_e = [\mathbf{e}_0, \ldots, \mathbf{e}_T] \in \mathbb{R}^{(T+1)\times d_G}$. To further capture contextual information of these vectors, we adopt the self-attention block [32, 42] to generate another $T+1$ vectors, which are finally aggregated with summation resulting in the d-dimensional vector $\hat{\mathbf{x}}_e$. In addition, we also represent each candidate attribute entity e_{T+1} as a vector $\mathbf{e}_{T+1}$ via the pretrained TransE embedding, and then map it into a d-dimensional space via a residual block [15] that consists of a feed forward network and residual connections, i.e., $\text{RB}(\mathbf{e}_{T+1}) = \xi(W_{r2}(\xi(W_{r1}\mathbf{e}_{T+1}) + \mathbf{e}_{T+1}))$, where $\xi(x) = \max(0, x)$ is the ReLU activation function and $W_{r_1} \in \mathbb{R}^{d_G \times d_G}$, $W_{r_2} \in \mathbb{R}^{d \times d_G}$ are learnable parameters.

Dialogue Representation. The T-turn dialogue is represented by a matrix $x_d \in |V|^{(2T+1)\times d_w}$, where V denotes the vocabulary, d_w is the maximum length of a question or a response, and each entry refers to a word index in V. We then feed x_d to a d_w-dimensional word embedding layer pretrained via word2vec [23] on the raw dialogue corpus, and denote the output as a tensor $\mathbf{x}_d \in \mathbb{R}^{(2T+1)\times L \times d_w}$. Since the next question is more likely to be related to the most recent turns, we compute the average of the latest T_0 utterances and obtain $\bar{\mathbf{x}}_d = \frac{1}{T_0}\sum_{t=0}^{T_0-1} \mathbf{x}_{d,2T+1-j} \in \mathbb{R}^{L\times d_w}$. We adopt the same self-attention block as above (but without sharing weights) to encode the sequence of words. Mean-pooling is finally adopted to derive the final d-dimensional vector $\hat{\mathbf{x}}_d$. For each candidate question q_{T+1}, we also encode it with the same word embedding, self-attention block, and mean-pooling.



(a) Cellphone (b) Automotive

Figure 2: Human validation of question order

User and Item Representation. To model user–item interactions, we embed user $u \in \mathcal{U}$ and candidate item $v \in \mathcal{V}$ as d-dimensional vectors, denoted by $\mathbf{x}_{u,1}, \mathbf{y}_{v,1} \in \mathbb{R}^d$, which are regarded as the first part of the user and item representation. However, such encodings fail to account for historical user behavior and descriptive attributes of items, which are both captured by the graph. Therefore, we again leverage the pretrained TransE model from the previous section. Let $\mathbf{x}_{u,G}, \mathbf{y}_{i,G} \in \mathbb{R}^{d_G}$ denote the graph embedding of the user and the item, respectively. We then adopt a residual block to derive the second part of the user and item representation as $\mathbf{x}_{u,2} = \text{RB}(\mathbf{x}_{u,G} + \mathbf{r}_{ui})$ and $\mathbf{y}_{i,2} = \text{RB}(\mathbf{y}_{i,G})$, where $\mathbf{r}_{ui}$ denotes the relation embedding for the user–item interaction. By combining both parts, the final encoding of user and item becomes $\hat{\mathbf{x}}_u = [\mathbf{x}_{u,1}; \mathbf{x}_{u,2}]$ and $\hat{\mathbf{y}}_i = [\mathbf{y}_{i,1}; \mathbf{y}_{i,2}]$, respectively.

Attribute Entity Prediction. We draw on the dialogue history and previous attribute entities to predict the next one. We learn a function $f_{\text{nda}}(x_d, x_e, e_{T+1})$ to emit a score measuring the similarity between the dialogue x_d, attribute entity x_e, and a candidate entity e_{T+1}. It is defined as $f_{\text{nda}}(x_d, x_e, y_e) = \text{FF}([\hat{\mathbf{x}}_d; \hat{\mathbf{y}}_e]) + \text{FF}([\hat{\mathbf{x}}_e; \hat{\mathbf{y}}_e])$, where FF refers to a feedforward network and the corresponding objective is:

$$\mathcal{L}_{\text{nda}} = \mathbb{E}_{y_e^-}[\ell(f_{\text{nda}}(x_d, x_e, y_e), f_{\text{nda}}(x_d, x_e, y_e^-))], \tag{1}$$

Here, y_e^- denotes a negative sample of a descriptive attribute that is irrelevant to the chosen item, and $\ell(x^+, x^-) = -\log\sigma(x^+ - x^-)$ is the pairwise ranking loss, where $\sigma(\cdot)$ is the sigmoid function.

Next Question Prediction. We mainly rely on the dialogue history and descriptive attributes to select the next question to ask. Specifically, we aim to learn a scoring function $f_{\text{dial}}(x_d, x_e, q_{T+1})$ that estimates the similarity between the dialogue x_d, entities x_e, and a candidate question q_{T+1}. The function is defined as $f_{\text{dial}}(x_d, x_e, y_r) = \text{FF}([\hat{\mathbf{x}}_d; \hat{\mathbf{y}}_r]) + \text{FF}([\hat{\mathbf{x}}_e; \hat{\mathbf{y}}_r])$. Thus, for each dialogue in the training set, we can minimize the objective

$$\mathcal{L}_{\text{dial}} = \mathbb{E}_{q'}[\ell(f_{\text{dial}}(x_d, x_e, q_{T+1}), f_{\text{dial}}(x_d, x_e, q'))], \tag{2}$$

where q' denotes a negative sample of a wrong question.

Recommendation Prediction. Given the user x_u, candidate item y_i, and dialogue x_d, as well as descriptive attributes x_e, the recommender denoted by $f_{\text{rec}}(x_u, y_i, x_d, x_e)$ outputs a score indicating how well the candidate item matches the user given the dialogue context. We define f_{rec} as

$$f_{\text{rec}}(x_u, y_i, x_d, x_e) = \hat{\mathbf{x}}_u^\intercal \hat{\mathbf{y}}_i + \mathbf{W}_1[\hat{\mathbf{x}}_d; \mathbf{y}_{i,1}] + \mathbf{W}_2[\hat{\mathbf{x}}_e; \mathbf{y}_{i,1}], \tag{3}$$

where $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{2d}$ are learnable parameters. Therefore, for each dialogue consisting of (x_u, x_d, x_e, y_i) in the training set, we aim to minimize the objective

$$\mathcal{L}_{\text{rec}} = \mathbb{E}_{y_i^-}[\ell(f_{\text{rec}}(x_u, x_d, x_e, y_i), f_{\text{rec}}(x_u, x_d, x_e, y_i^-))], \tag{4}$$

where y_i^- denotes items that the user x_u never interacts with in the training set.

Objective We jointly learn three scoring functions across all training dialogue data by minimizing the overall joint objective $\mathcal{L} = \sum_{D_{\text{train}}} \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{dial}} + \mathcal{L}_{\text{nda}}$, where D_{train} denotes the training set of all input–output pairs.

For model parameters in the recommender f_{rec}, including user and item embeddings, we adopt SGD for optimization with a learning rate of 10^{-3} and weight decay of 10^{-4}. For the remaining parameters, we rely on Adam optimization with a learning rate of 5×10^{-4} and weight decay of 10^2. Since two optimizers may converge with different speeds, we make Adam backpropagate gradients every other epoch, while SGD updates across all 10 epochs. We set the batch size to 256. For model parameters, the sizes of word embedding, KG embedding, and user/item embedding are 200, 100, 100, respectively, and the latent vector dimensionality is $d = 100$. The multi-head attention size in the Transformer encoder is set to 4.

5 EXPERIMENTS

We extensively evaluate the proposed benchmark method over the HOOPS benchmark data. First of all, the model should be able to accurately conduct the next attribute prediction within the graph, to demonstrate the capability of pruning off irrelevant candidates within the HitL graph reasoning paradigm. Moreover, we expect the proposed HitL conversational recommendation to not only facilitate offering accurate recommendations but also to properly select the next questions to ask with user feedback, which correspond to the recommendation task and the next question prediction tasks, respectively. For each of these tasks, we compare our model against several state-of-the-art baselines.

Experimental Settings. Recall that our HOOPS dataset includes Cellphones & Accessories, Grocery & Gourmet, Toys & Games, and Automotive. Each provides a unique KG and a set of conversations, implying that results are not necessarily comparable across different domains. We split the conversations into training (60%), validation (20%), test (20%) portions. For each user–item pair, we take one conversation with a maximum utterance length of 50 and a maximum conversation length of 10, applying zero-padding if the number of utterances is less than 10. There are 10 question candidates to predict, out of which only one is the correct ground truth choice. The same setup also applies for next-hop entity prediction. For recommendation, we sampled 100 items with which the user has not interacted as negative candidates. Our goal is to retrieve 1 correct labeled item out of a pool of 100 candidates, 1 question out of 10 question candidates, as well as 1 entity out of 10.

Baselines. For recommendation task, we consider Bayesian personalized ranking **BPR** [29], collaborative knowledge base embedding **CKE** [43], **RippleNet** [33], and the knowledge graph attention network **KGAT** [35] as baselines. For next-question prediction, we compare popular response ranking methods, including the deep matching network **DMN** [40], deep attention matching network **DAM** [48], and multi-hop selector network **MSN** [48]. The baselines above each either yield recommendations or address the next question prediction task. However, none of them is able to accommodate both tasks. Hence, we implement the following modified baselines targeted at jointly conducting both tasks. **KBRD** [7]: This is a conversational recommender system that originally couples recommendation with dialogue generation. We applied Transformers [32] with a decoder designed for our response selection downstream task. **OpenDialKG** [24]: The DialKG Walker model is able to conduct conversational reasoning. The original version supports predicting a KG entity via an attention-based graph path decoder. We modified the model by encoding the target question with an LSTM, which enables next question prediction.

Next Attribute Prediction. We study the performance of descriptive attribute prediction to justify whether the HitL graph reasoning is able to correctly predict the next attribute entity. Since the KG incorporates meta-information of both users and items, predicting the most relevant entities manifests a proper user participation that enables pruning off irrelevant candidates. The results in Table 2 indicate that our baseline approach obtains the best results compared to all prior baselines. Seq2Seq and LSTM are typical methods designed for sequential prediction, but they are unable to perform well with the aid of graph structures. Moon et al. [24] deployed a graph decoder by walking over knowledge graphs. However, without considering the hybrid user behavior in the modeling, it remains less convincing in terms of the transparency.

Next Question Prediction. In our benchmark dataset, we assume users may occasionally struggle to provide useful requests to the agent, since they initially may not be entirely aware of their preferences. Thus, learning to ask the right question given the past conversation context reveals whether the model successfully predicts user preferences. The benchmark results are shown in Table 2. In our HitL graph reasoning for conversational recommendation scenario, next question prediction closely resembles response ranking. The OpenDialKG and KBRD baselines exploit KGs in order to leverage sentence, dialogue, and KG structural features. Our proposed benchmark method not only takes advantage of the extracted coarse-to-fine entities within the KG, but also models the user feedback within the conversational turns. This enables it to outperform other baselines in most of the evaluation results.

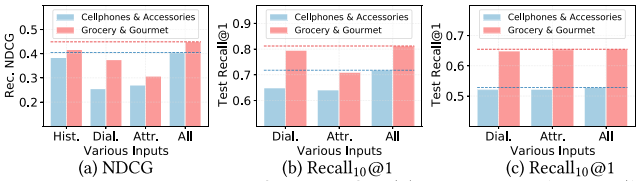
Recommendation. We adopt standard metrics to evaluate the recommendations of each user in the testset, including Normalized Discounted Cumulative Gain (**NDCG**), **Recall**, and Mean Average Precision (**MAP**). The top-10 recommendation results of different models are given in Table 2. The benchmark method is able to outperform other approaches, as it draws on human feedback and HitL graph reasoning to enhance the recommendation quality.

Ablation Study. We show the influence of different modules taking care of corresponding inputs on the three sub-tasks to demonstrate the effectiveness of our designed framework. As shown in Figure 3(a), we first consider the recommendation performance with each input separately with abbreviations Hist. = User History, Dial. = dialogue, and Attr. = descriptive attributes. While keeping all other parameters unchanged, we observe that each input contributes substantially to the performance, but retaining only one of them leads to a performance drop. This suggests that each ingredient of our HitL approach is complementary rather than redundant. The model is almost equal to user-based collaborative filtering when the input is solely user behavior, which takes the dominant role for personalized recommendation. In contrast, although the dialogue provides more semantics than pure attributes, it is worth noting that the conversational utterances may also introduce noise in the input. Therefore, there is a slight recommendation performance gap between dialogue-alone and attribute-alone as input.

Furthermore, we also evaluate how the various inputs contribute to the next question prediction and next attribute prediction subtasks in Figures 3(b) and (c). We find that user-readable dialogue is

Tasks	Benchmarks	Cellphones & Accessories			Grocery & Gourmet			Toys & Games			Automotive		
Next Attribute	Metrics	MAP	R_{10}@1	R_{10}@3	MAP	R_{10}@1	R_{10}@3	MAP	R_{10}@1	R_{10}@3	MAP	R_{10}@1	R_{10}@3
	Seq2Seq	0.612	0.430	0.738	0.707	0.544	0.845	0.593	0.431	0.674	0.701	0.547	0.817
	LSTM	0.642	0.465	0.772	0.726	0.569	0.859	0.566	0.408	0.626	0.659	0.499	0.768
	OpenDialKG	0.643	0.467	0.774	0.707	0.555	0.822	0.656	0.501	0.754	0.706	0.557	**0.838**
	HOOPS (Ours)	**0.688**	**0.528**	**0.810**	**0.789**	**0.655**	**0.917**	**0.705**	**0.561**	**0.806**	**0.712**	**0.564**	0.825
Next Question	Metrics	MAP	R_{10}@1	R_{10}@3	MAP	R_{10}@1	R_{10}@3	MAP	R_{10}@1	R_{10}@3	MAP	R_{10}@1	R_{10}@3
	DMN [40]	0.475	0.269	0.564	0.502	0.304	0.587	0.456	0.253	0.518	0.469	0.267	0.553
	DAM [48]	0.514	0.373	0.590	0.581	0.394	0.635	0.579	0.388	0.546	0.552	0.387	0.608
	MSN [42]	0.608	0.428	0.740	0.678	0.503	0.749	0.630	0.455	0.732	0.645	0.473	0.713
	OpenDialKG [24]	0.699	0.654	0.678	0.729	0.676	0.724	0.579	0.499	0.561	0.710	0.640	0.726
	KBRD [7]	0.669	0.498	0.771	0.768	0.626	**0.896**	0.688	0.559	**0.760**	0.711	0.552	0.809
	HOOPS (ours)	**0.781**	**0.718**	**0.788**	**0.854**	**0.812**	0.859	**0.693**	**0.562**	0.746	**0.850**	**0.805**	**0.858**
Recommend	Metrics	NDCG	Recall	MAP	NDCG	Recall	MAP	NDCG	Recall	MAP	NDCG	Recall	MAP
	BPR [29]	0.349	0.540	0.336	0.331	0.521	0.360	0.305	0.498	0.335	0.307	0.487	0.312
	CKE [43]	0.360	0.543	0.303	0.411	0.598	0.353	0.435	0.636	0.372	0.385	0.570	0.327
	RippleNet [33]	0.326	0.476	0.279	0.366	0.534	0.314	0.420	0.612	0.361	–	–	–
	HeteroEmbed [1]	0.388	0.583	0.327	0.439	0.637	0.377	0.467	0.654	0.409	0.395	0.598	0.335
	KGAT [35]	0.399	0.593	0.338	0.424	0.622	0.363	0.443	0.637	0.386	0.387	0.581	0.326
	KBRD [7]	0.253	0.424	0.201	0.293	0.475	0.237	0.210	0.366	0.162	0.249	0.409	0.200
	HOOPS (ours)	**0.405**	**0.611**	**0.341**	**0.449**	**0.650**	**0.386**	**0.477**	**0.668**	**0.418**	**0.403**	**0.605**	**0.341**

Table 2: Performance of selected baselines and our benchmark methods on four proposed sub-datasets. The best results are highlighted in bold and the second best results are underlined.



(a) NDCG (b) Recall$_{10}$@1 (c) Recall$_{10}$@1

Figure 3: Comparison of inputs for (a) recommendation, (b) next question prediction, and (c) next attribute prediction.

more useful than merely considering the attributes for the question prediction task. Interestingly, there is a small performance gap for next attribute prediction. This is also because the utterance incorporates the descriptive attribute information, while attribute-alone loses semantic information content. Thus, utterance-alone is better than attribute-alone on question prediction, but fairly similar on the attribute prediction sub-task.

6 RELATED WORK

There has been significant research in human-centered AI. Much of it has focused on societal goals rather than individual human needs and interests [16, 17, 21]. Recently, some progress has been made in the HCI field towards invoking ML to augment interactive and intelligent systems [2, 41]. In this regard, the notion of Human-in-the-Loop (HitL) AI has been proposed. We propose a concrete HiTL graph reasoning framework for conversational recommendation. At the same time, the integration of knowledge graphs [20] has enabled CRS models to make recommendations grounded in knowledge-driven reasoning [11, 18, 24, 37, 38]. For example, Lei et al. [18] propose an RL-based mechanism based on an interactive path reasoning algorithm. However, the lack of human-readable fluent utterances is replaced by crawling the attribute words from raw review contexts, which is less practical in real-world scenarios. In Chen et al. [7], item-related knowledge bases with entity-linked text leads to better performance than either of them alone in dialogue generation and recommendation. Comparing to these methods, we provide an open dataset for conversational recommendation that supports the HiTL graph reasoning paradigm and integrates knowledge graphs so that prominent knowledge with semantics can be used to consider user-involved feedback and provide transparent recommendations. Except for conversational recommendation, the dataset may also be used for conversational search [3], conversational QA [28] and Explainable Recommendation [8, 9, 19, 44, 46]. Since reasoning on graphs naturally provides transparency of the decision making process, it helps to provide explanations for users over the recommended items [1, 11, 38, 39].

7 CONCLUSION

Our work in this paper is the first exploration of human-in-the-loop (HitL) learning for recommendation. Specifically, we define a new HitL graph reasoning paradigm with the three properties of hybrid integration, coarse-to-fine resolution, and a transparent decision-making process. We instantiate the paradigm for the conversational recommendation problem, where the system can leverage interactive user feedback to shrink the large search space during the multi-step reasoning process. Accordingly, we construct a new dataset called HOOPS including a graph that structurally integrates diverse user behavior and item-related information, as well as a multi-round conversation corpus that simulates user–agent interaction. We also provide a benchmark model to approach the HitL graph reasoning for recommendation with reported performance in three tasks on the constructed dataset. We hope it opens up avenues for further research on more realistic applications for Human-in-the-Loop learning. All data and code are freely available under a CC-BY-SA license.[1]

ACKNOWLEDGEMENT

This work was supported in part by NSF IIS-1910154 and IIS-2007907. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsors.

[1] https://github.com/zuohuif/HOOPS

REFERENCES

[1] Q. Ai, V. Azizi, X. Chen, and Y. Zhang. 2018. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms* (2018).

[2] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. Bennett, K. Quinn, J. Teevan, R. Kikin-Gil, and E. Horvitz. 2019. Guidelines for Human-AI Interaction. *CHI* (2019).

[3] Keping Bi, Qingyao Ai, Yongfeng Zhang, and W Bruce Croft. 2019. Conversational product search based on negative feedback. In *CIKM*.

[4] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NeurIPS*.

[5] P. Budzianowski, T. Wen, B. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gasic. 2018. MultiWOZ - A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling. In *EMNLP*.

[6] Hanxiong Chen, Shaoyun Shi, Yunqi Li, and Yongfeng Zhang. 2021. Neural Collaborative Reasoning. *WWW* (2021).

[7] Q. Chen, J. Lin, Y. Zhang, M. Ding, Y. Cen, H. Yang, and J. Tang. 2019. Towards Knowledge-Based Recommender Dialog System. *ArXiv 1908.05391* (2019).

[8] Xu Chen, Hanxiong Chen, Hongteng Xu, Yongfeng Zhang, Yixin Cao, Zheng Qin, and Hongyuan Zha. 2019. Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation. In *SIGIR*.

[9] Xu Chen, Yongfeng Zhang, and Zheng Qin. 2019. Dynamic explainable recommendation based on neural attentive models. In *AAAI*.

[10] M. Eric, L. Krishnan, F. Charette, and C. D. Manning. 2017. Key-Value Retrieval Networks for Task-Oriented Dialogue. In *SIGDIAL*.

[11] Zuohui Fu, Yikun Xian, Ruoyuan Gao, Jieyu Zhao, Qiaoying Huang, Yingqiang Ge, Shuyuan Xu, Shijie Geng, Chirag Shah, Yongfeng Zhang, et al. 2020. Fairness-aware explainable recommendation over knowledge graphs. In *SIGIR*. 69–78.

[12] Zuohui Fu, Yikun Xian, Yongfeng Zhang, and Yi Zhang. 2020. Tutorial on Conversational Recommendation Systems. In *RecSys*.

[13] Zuohui Fu, Yikun Xian, Yongfeng Zhang, and Yi Zhang. 2021. IUI 2021 Tutorial on Conversational Recommendation Systems. In *IUI*.

[14] Zuohui Fu, Yikun Xian, Yongfeng Zhang, and Yi Zhang. 2021. WSDM 2021 Tutorial on Conversational Recommendation Systems. In *WSDM*.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

[16] A. Jaimes, D. G. Perez, N. Sebe, and T. Huang. 2007. Guest Editors' Introduction: Human-Centered Computing–Toward a Human Revolution. *Computer* (2007).

[17] R. Kling and S. L. Star. 1998. Human centered systems in the perspective of organizational and social informatics. *SIGCAS Comput. Soc.* 28 (1998), 22–29.

[18] Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020. Interactive Path Reasoning on Graph for Conversational Recommendation. In *KDD*.

[19] Lei Li, Yongfeng Zhang, and Li Chen. 2020. Generate neural template explanations for recommendation. In *CIKM*.

[20] Zelong Li, Jianchao Ji, Zuohui Fu, Yingqiang Ge, Shuyuan Xu, Chong Chen, and Yongfeng Zhang. 2021. Efficient Non-Sampling Knowledge Graph Embedding. *WWW* (2021).

[21] Zhenyu Liao, Yikun Xian, Xiao Yang, Qinpei Zhao, Chenxi Zhang, and Jiangfeng Li. 2018. TSCSet: A crowdsourced time-sync comment dataset for exploration of user experience improvement. In *IUI*. 641–652.

[22] Michael E. J. Masson. 1983. Conceptual processing of text during skimming and rapid sequential reading. *Memory & Cognition* 11 (1983), 262–274.

[23] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NeurIPS*.

[24] Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. OpenDialKG: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *ACL*.

[25] Mark Newman. 2010. Networks: An Introduction.

[26] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In *EMNLP*.

[27] Namyong Park, Andrey Kan, Xin Dong, Tong Ke Zhao, and Christos Faloutsos. 2019. Estimating Node Importance in Knowledge Graphs Using Graph Neural Networks. *KDD* (2019).

[28] Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W Bruce Croft, and Mohit Iyyer. 2019. Attentive history selection for conversational question answering. In *CIKM*.

[29] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*.

[30] Shaoyun Shi, Hanxiong Chen, Weizhi Ma, Jiaxin Mao, Min Zhang, and Yongfeng Zhang. 2020. Neural Logic Reasoning. In *CIKM*. 1365–1374.

[31] Yueming Sun and Yi Zhang. 2018. Conversational Recommender System. In *SIGIR '18*.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. 2017. Attention is All you Need. In *NIPS*.

[33] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *CIKM*. ACM, 417–426.

[34] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge Graph Convolutional Networks for Recommender Systems. In *WWW '19*.

[35] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *KDD*.

[36] Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2018. Learning Neural Templates for Text Generation. In *EMNLP*.

[37] Yikun Xian, Zuohui Fu, Qiaoying Huang, Shan Muthukrishnan, and Yongfeng Zhang. 2020. Neural-Symbolic Reasoning over Knowledge Graph for Multi-Stage Explainable Recommendation. *arXiv preprint arXiv:2007.13207* (2020).

[38] Yikun Xian, Zuohui Fu, S. Muthukrishnan, Gerard de Melo, and Yongfeng Zhang. 2019. Reinforcement Knowledge Graph Reasoning for Explainable Recommendation. *SIGIR* (2019).

[39] Yikun Xian, Zuohui Fu, Handong Zhao, Yingqiang Ge, Xu Chen, Qiaoying Huang, Shijie Geng, Zhou Qin, Gerard De Melo, Shan Muthukrishnan, et al. 2020. CAFE: Coarse-to-fine neural symbolic reasoning for explainable recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 1645–1654.

[40] L. Yang, M. Qiu, C. Qu, J. Guo, Y. Zhang, W. B. Croft, J. Huang, and H. Chen. 2018. Response Ranking with Deep Matching Networks and External Knowledge in Information-seeking Conversation Systems. *SIGIR* (2018).

[41] Qian Yang, Nikola Banovic, and John Zimmerman. 2018. Mapping Machine Learning Advances from HCI Research to Reveal Starting Places for Design Innovation. *CHI* (2018).

[42] Chunyuan Yuan, Wen jie Zhou, MingMing Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. 2019. Multi-hop Selector Network for Multi-turn Response Selection in Retrieval-based Chatbots. In *EMNLP/IJCNLP*.

[43] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative Knowledge Base Embedding for Recommender Systems. In *KDD*.

[44] Yongfeng Zhang and Xu Chen. 2018. Explainable Recommendation: A Survey and New Perspectives. *Found. Trends Inf. Retr.* 14 (2018), 1–101.

[45] 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*. ACM, 177–186.

[46] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 2014. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 83–92.

[47] Wayne Xin Zhao, Gaole He, Kunlin Yang, Hongjian Dou, Jin Huang, Siqi Ouyang, and Ji-Rong Wen. 2019. KB4Rec: A Data Set for Linking Knowledge Bases with Recommender Systems. *Data Intelligence* 1, 2 (2019), 121–136.

[48] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-Turn Response Selection for Chatbots with Deep Attention Matching Network. In *ACL*.

[49] Yaxin Zhu, Yikun Xian, Zuohui Fu, Gerard de Melo, and Yongfeng Zhang. 2021. Faithfully Explainable Recommendation via Neural Logic Reasoning. *NAACL* (2021).