Keyword-guided Topic-oriented Conversational Recommender System

Yiming Pan
College of Computer Science
Chongqing University
Chongqing 400044, China
panyiming@cqu.edu.cn

Yunfei Yin*
College of Computer Science
Chongqing University
Chongqing 400044, China
yinyunfei@cqu.edu.cn

Faliang Huang*
Guangxi Key Lab of Human-machine
Interaction and Intelligent Decision
Nanning Normal University
Nanning 530100, China
faliang.huang@gmail.com

2022 International Joint Conference on Neural Networks (IJCNN) | 978-1-7281-8671-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/IJCNN55064.2022.9892470

***Abstract*—Conversational recommender system (CRS) allows agent to understand the conversation with user and give recommendations after multi-turn dialogues. However, there are still two limitations in existing CRS: (1) improper words or items may be chosen for the given topic in the generation, and (2) the contextual information of items in the recommendation is not rationally explored. To solve these issues, we proposed a Keyword-guided Topic-oriented CRS model (KGTO), which captures more accurate topic by extracting keywords through the hierarchical attention mechanism, and enriches the contextual information of items by fusing the co-occurrence graph with the knowledge graph. Moreover, a generative module can select words or items supplemented topic information to generate proper responses. Extensive experiments on the task-oriented dialogue dataset prove that our model performs well in recommendation effectiveness and dialogue informativeness.**

***Index Terms*—Conversational Recommender System, Dialogue Geeneration, Topic Model, Dialogue System**

I. Introduction

The CRS can capture the information of history dialogue from the conversation, learn a high-quality item representation, and match the dialogue with items to get good recommendation results [1]–[3]. In the setting of travel conversations, the agent should be able to capture latent topics that changes over time, such as booking hotels, finding attractions, reserving restaurants, etc. And it can capture keyword information under corresponding topics, such as location and rating in the topic of finding attractions, to recommend items related to topics. However, how to create a unified model that combines conversation and recommendation still remains challenging.

One problem is that the latent topic information cannot be accurately captured. The latent topic information is helpful for narrowing the candidate word set or item set in response generation, and it is able to assist agent to select topic-related words or items and keep the topic consistency of the conversation. Previous studies lack of capturing the latent topic [2], [4], such as "Booking Hotels", "Finding Attractions" in Figure 1, which causes some words or items unrelated to the topic to be chosen and generate irrelevant responses. Besides, [1] uses all the words appearing in the conversation to learn the latent topic. It lacks of filtering out uninformative words such as stopwords which makes the learned latent topic inaccurate compared with the actual topic. If we focus on the keyword, such as "swimming pool", "game hall", "cheap", "4 stars", etc. in Figure 1, we will learn an accurate topic information, and the keyword information can be used as the explainable information of the item to generate informative responses. Therefore, we learn the latent topic by extracting keywords through hierarchical attention mechanism to guide the response generation.

Another challenge lies in not fully exploring the relationship between items and words. Since each item is always associated with some words in the conversation, we can explore the relationship between them to let the item contain more contextual information. As shown in Figure 1, the item "Navarro Hotel" is associated with the words "priced", "guest", "house", while "San Antonio Sea World" is related to "world", "famous", "sea world". However, previous works [1], [3], [5] only consider the relationship between items or the relationship between non-item words, which may cause the contextual information to be lost for item representations and the recommendation information to be lost for word representations, and finally produce low-quality representations.

To address these issues, we proposed a generative CRS model, called Keyword-guided Topic-oriented Conversational Recommender System (KGTO), where the keyword information in the conversation is extracted for learning latent topic information, and the learned latent topic information is further used to guide the selection of items for recommendations and words for response generation. In addition, the co-occurrence graph and the knowledge graph are fused to enrich the contextual information of items, and then each item is matched with history dialogue to get a better recommendation.

Our contributions are summarized below: (1) A hierarchical attention mechanism is proposed to capture global topic information, and facilitate the selection of words and items in the response; (2) In order to obtain a higher-quality item representation, a graph fusion strategy is designed to integrate the item-word co-occurrence with the item knowledge; (3) Extensive experiments on a task-oriented dialogue dataset

Supported by the Natural Science Foundation of China under Grant 61962038, and by the Guangxi Bagui Teams for Innovation and Research under Grant 201979.

*Corresponding author

indicate that, KGTO performs satisfactory in terms of both automatic evaluations and human evaluations.

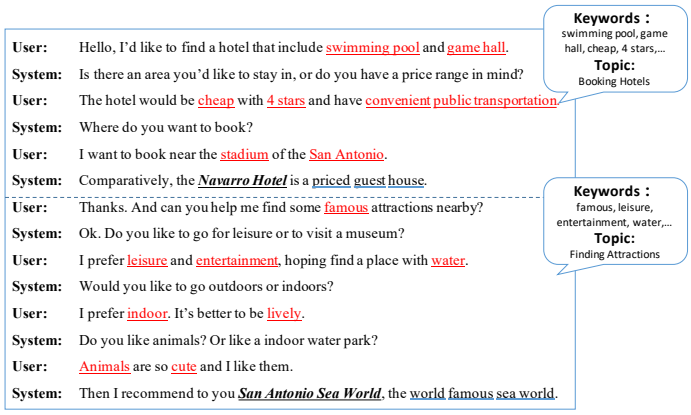

Fig. 1. An example of a conversation with topics and keywords.

II. Related Works

With the application of voice assistants and shopping assistants in commercial area, the conversational recommender system has attracted much attention in intelligent dialogue. Instead of providing all requirements in one step in traditional recommender system, the CRS can converse with users to implement recommendations using natural language in a cold-start setting, which can guide users to express their preferences through task-oriented and multi-turn dialogues, and infer the dynamic preferences of users from the conversation to make proper recommendations [6]–[8]. Previous works on CRS can be roughly categorized into two types:

One is attribute-based CRS [4], [9]–[11] that captures user preferences by asking the user about attributes of items and uses pre-defined templates to generate responses. [9], [11]–[13] apply reinforcement learning model, and learn history dialogue by facet-value pairs [14], but it requires manual settings and its generalization ability is insufficient. [11] represents items by FM [15], which only considers the relationship between items. [10] sets up a trigger to recommend items when the confidence is high, but lacks the module to capture rich contextual and semantic information. [4] considers topic information with the supervised training in the conversation, which can understand the conversation better in the global.

The other is generation-based CRS [2], [3], [5], [16], [17], which pays attention to generate human-like responses in natural language. [1], [5] learn latent topic distribution from history dialogue and generate responses relevant to the topic, but they lack a unified model to capture local keyword information and global topic information in history dialogue, and do not let the topic information guide the selection of non-item words and items. [16] improves the performance of recommendation system by introducing knowledge-based user preference information in the conversation system, and applies Transformer [18] to learn the representation of history dialogue, but it lacks of capturing the global topic information to understand the history dialogue better. Previous works assumed that the topics discussed by users would not change over time, such as only discussing movies, while hot topics in social media would change over time, not just guided by a single topic. [5] proposed a dynamic CRS that captures topics or user interests from the perspective of time and explores settings where topics change over time, but it does not use topic information to generate the response. [19] provides a topic-annotated CRS dataset, which can generate responses under the guidance of the topic, but not all conversations have the topic annotation, and the ability of generation is insufficient. Besides, our model does not rely on annotated topic information, but uses unsupervised leaning to learn the topic information in the conversation. [3], [20], [21] incorporate both word-oriented and item-oriented knowledge graphs to align the word-level and item-level semantic spaces based on the mutual information maximization. However, they only use external knowledge information to explore the relationship between items and words, and lack co-occurrence analysis of items and words in the conversation, which is unable to fully integrate contextual information into recommended items. In addition, topic information is essential for generating responses related to the topic to maintain the consistency of the conversation, but it lacks the guarantee of topic consistency. Our model integrates the information of conversation and recommendation into generative language model to get system responses, so our model belongs to generation-based CRS.

III. The Proposed Model

Our model consists of four modules: (1) History Dialogue Representation, which learns history dialogue representation between user and system; (2) Word/Item Representation, which learns the representation of words and items; (3) Topic Representation, which extracts the topic information from history dialogue; (4) Response Generation, which generates an appropriate response according to the learned dialogue representation and word/item representation, supplemented by topic information. Figure 2 shows the model overview.

A. History Dialogue Representation

In the CRS, an informative representation of history dialogue is useful for recommendation, especially under the cold-start setting. Inspired by [22]–[24], we design a two-layer attention mechanism, which is expatiated in this section, to put focus more on important information when learning the representation of history dialogue.

1) Word-level Attention: Given an utterance s_i with L_i words $\left\{x_1^i, x_2^i, ..., x_{L_i}^i\right\}$, we first transform each word x_t^i into a real-value vector w_t^i:

$$w_t^i = Embedding\left(x_t^i\right) \tag{1}$$

And then we employ Bi-GRU model to generate contextual hidden vector h_t^i of each word:

$$h_t^i = BiGRU\left(w_t^i\right), i \in [1, T], t \in [1, L_i] \tag{2}$$

which summarizes the information of the whole utterance centered around each word. But not each word can provide useful

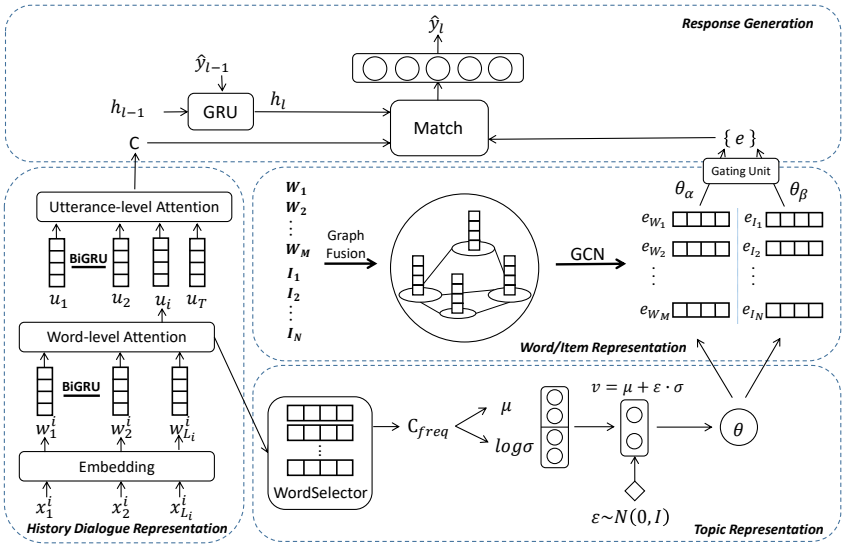

Fig. 2. Overall structure of our model.

information for the utterance representation, so we design a word-level attention mechanism to extract such words that are important to the meaning of the utterance and aggregate the representation of those informative words to form a good utterance representation. Firstly, we feed hidden state h_t^i from Bi-GRU through a one-layer MLP to get the word hidden vector $h_t^{i\prime}$:

$$h_t^{i\prime} = tanh\left(Wh_t^i + b\right) \tag{3}$$

and then we get a normalized importance weight α_t^i through softmax and ReLU function:

$$\alpha_t^i = ReLU\left(\frac{exp\left(h_t^{i\prime}\right)}{\sum exp\left(h_t^{i\prime}\right)}\right) \tag{4}$$

and we compute the utterance vector u_i by using a weighted sum of word hidden vectors based on the weights:

$$u_i = \sum \alpha_t^i h_t^i \tag{5}$$

2) Utterance-level Attention: Given a history dialogue sequence $\{u_1, u_2, ..., u_T\}$ which contains T utterances, for each utterance, we take the vector u_i obtained by word-level attention as the representation of each utterance. And then, we learn utterance representation h_i with contextual information through Bi-GRU:

$$h_i = BiGRU\left(u_i\right), i \in [1, T] \tag{6}$$

The Bi-GRU contains the forward GRU which reads the dialogue from u_1 to u_T and a backward GRU which reads the dialogue from u_T to u_1 to get the representation of each utterance, which summarizes the information of the whole dialogue centered around each utterance. Similar to word-level attention, at the utterance level, not each utterance provides useful information for the representation of history dialogue. We apply an utterance-level attention mechanism to learn history dialogue. Firstly, we feed hidden state h_i from Bi-GRU through a one-layer MLP to get the utterance hidden vector h_i':

$$h_i' = \tanh\left(Wh_i + b\right) \tag{7}$$

then we get a normalized importance weight α_i through softmax and ReLU function:

$$\alpha_i = \mathrm{ReLU}\left(\frac{\exp\left(h_i'\right)}{\sum \exp\left(h_i'\right)}\right) \tag{8}$$

and we compute the history dialogue vector C by using a weighted sum of utterance hidden vectors based on the weights:

$$C = \sum \alpha_i h_i \tag{9}$$

B. Word/Item Representation

1) Fusion of Co-occurrence Graph and Knowledge Graph: In the conversation, each item is always associated with one or a few specific words. For example, the recommended item "San Antonio Sea World" co-occurs with the word "world", "famous", "sea world". The external knowledge graph only contains the relationship between items and lacks the relationship between items and words. In order to enrich the contextual information of items and explore the relationship between items and words, we fuse the item-word co-occurrence graph with the item-item knowledge graph in the corpus, and form an undirected fused graph structure as $G = (N, E)$, where $N = \{W \cup I\}$ is a set of nodes and $E \subseteq N \times N$ is a set of edges between nodes. Here the node can be items like "restaurant" or "hotel" etc. from item set I, and words like "swimming pool" or "game hall" etc. from non-item word set W, while the edge can be "have" or "not have" etc. from E. The fusion algorithm is shown in Algorithm 1, where the input is a graph G_{know} between items with external knowledge and dialogue context in the corpus, and the output is the fused graph G. In this way, words and items can be connected closely.

Algorithm 1 GraphFusion

Input: $\mathbf{G}_{know}$, Context
Output: G

for item i_k in G_{know} **do**
  Locate i_k in the Context using NER
  Find keywords in the Context of i_k
  Add keywords as the neighbor nodes of i_k
end for
Add all keywords appearing in the neighbor nodes of items as the node of G_{know}
for keyword w_k in G_{know} **do**
  if w_k in the neighbor node of i_k **then**
    Add i_k as the neighbor node of w_k
  end if
end for

2) Graph Node Representation Learning: Recently, Graph Neural Networks like Graph Convolutional Networks (GCNs) [25] have played an important role in CRS [5], [26]. Unlike content-based deep models (e.g., RNNs), GCNs can generate a hidden representation of each node based on graph structure and content information. In this study, we utilize GCNs to encode the node representation $h^{(l)}$ in l^{th} layer:

$$h^{(l)} = \mathrm{Re}LU\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}h^{(l-1)}W^{(l)}\right) \tag{10}$$

where $h^{(l-1)}$ denotes the former layer node representation, $\widetilde{D}$ denotes the degree matrix, $\widetilde{A}$ denotes the adjacency matrix with added self-connections, and $W^{(l)}$ is weight matrix in l^{th} layer. At the final layer L, the GCN embedding h^L, which contains the embedding of words and items, is taken as the node representation e. According to this method, we can get the word representation containing item information and the item representation containing word information from e, which are denoted as $e_W = \{e_{W_1}, e_{W_2}, ..., e_{W_M}\}$ and $e_I = \{e_{I_1}, e_{I_2}, ..., e_{I_N}\}$ respectively.

C. Topic Representation

We aim to predict the topic of the next utterance to select words or items for response generation. Topic models [27]–[30] perform well in learning latent topic information, but these models cannot be applied to learn latent topic underlying in dialogues, since they do indiscriminately take all words rather than informative words in dialogues for topic extraction. Different from DCR selecting all the words in history dialogue, we apply a module called WordSelector, which can dynamically filter out words that are not helpful for learning the latent topic. We firstly extract informative words from the Word-level Attention based on weight of each word:

$$WordSelector\,(w) = \left\{w_t^i | \alpha_t^i > \lambda\right\} \tag{11}$$

where $i \in [1, T]$, $t \in [1, L_i]$ and λ is a threshold which is a learnable parameter.

History dialogue representation C is a continuous high-dimensional vector, which make the computation of marginal Gaussian distribution $p\,(\theta|C)$ very difficult. Here we adopt variational inference to estimate the topic information of history dialogue. For each word in W, it is assigned a value according to the times it appears in history dialogue, the value that appears k times is k, and the value that does not appear is 0. And we compose these values into a word-frequency vector C_{freq} according to the order in which they appear in W, but the words that do not appear in WordSelector are still assigned 0. Assuming that $q\,(\theta|C_{freq})$ is the variational distribution of marginalized variables. Inspired by neural variational inference model [31], we construct $q\,(\theta|C_{freq})$ as a feed-forward neural network for inference:

$$q\,(\theta|C_{freq}) = N\left(\theta; \mu\,(C_{freq}), \log\sigma\,(C_{freq})\right) \tag{12}$$

$$\mu\,(C_{freq}) = \mathrm{Re}LU\,(W_\mu g\,(C_{freq}) + a_\mu) \tag{13}$$

$$\log\sigma\,(C_{freq}) = \mathrm{Re}LU\,(W_\sigma g\,(C_{freq}) + a_\sigma) \tag{14}$$

where $g\,(\cdot)$ denotes the feed-forward neural network and weight matrices W_μ, W_σ and biases a_μ, a_σ are learnable parameters. Then we denote θ as topic distribution vector with Gaussian reparameterization trick [32]:

$$\theta = \mu + \varepsilon \cdot \sigma, \varepsilon \sim N\,(0, I) \tag{15}$$

where ε is the parameter of Gaussian distribution.

D. Response Generation

Different from the dialogue system only considering the word generation, the CRS needs to generate words and items simultaneously in the response. And the learned topic can be utilized to assistant the selection of words and items. Thus, we design a response generation module which matches words and items with learned latent topic to obtain the topic weight respectively, and select the words and items with high scores to generate topic-consistent responses.

1) Topic Weight Assignment: There are numerous non-item words and items in the candidate set, and how to accurately select non-item words or items is still a challenge. We match each non-item word or item with topic vector θ to get the non-item word weight θ_{W_i} and the item weight θ_{I_i}:

$$\theta_{W_i} = \sigma\,(\theta \cdot e_{W_i}) \tag{16}$$

$$\theta_{I_i} = \sigma\,(\theta \cdot e_{I_i}) \tag{17}$$

In this way, the non-item words and items related to the topic information will be given higher weights.

2) Word/Item Prediction and Response Generation: Choosing proper tokens is beneficial to generate high-quality responses. Given previous $l-1$ words, we first update the hidden state h_l based on GRU at timestamp l:

$$h_l = GRU\,(h_{l-1}, \widehat{y}_{l-1}) \tag{18}$$

where h_{l-1} and $\widehat{y}_{l-1}1$) are the hidden state and the predicted token at the timestamp $l-1$ respectively. Then, we combine

h_l with history dialogue representation C to match each non-item word or item with topic weight information to get the non-item word score or the item score:

$$\widehat{s}_W = \sigma\left((h_l \oplus C) \cdot (\theta_{W_i} \cdot e_{W_i})\right) \tag{19}$$

$$\widehat{s}_I = \sigma\left((h_l \oplus C) \cdot (\theta_{I_i} \cdot e_{I_i})\right) \tag{20}$$

where $\oplus$ is an operation concatenating h_l and C. And then we select the token with the highest score as the next generated token $\widehat{y}_l$ based on a gating unit like DCR. Finally, the system response u_{T+1} can be formed as follows:

$$u_{T+1} = \{\widehat{y}_1, \widehat{y}_2, ..., \widehat{y}_L\} \tag{21}$$

We can quickly choose some non-item words or items related to topic information, in order to generate more coherent system responses.

E. Model Optimization

For topic prediction, we use Kullback-Leibler Divergence [33] to calculate the asymmetry measure of difference between the assumed Gaussian distribution $N(0, I)$ and the latent topic distribution $q(\theta|C_{fre})$:

$$L_{topic} = -D_{KL}\left(N(0, I) \| q(\theta|C_{fre})\right) \tag{22}$$

where I is hyperparameter of the Gaussian distribution. For word and item prediction, we suppose that the real words and items are y_w and y_i respectively. The predicted words and items are $\widehat{y}_w$ and $\widehat{y}_i$ respectively. We use two loss functions to calculate the word loss L_w and the item loss L_i:

$$L_w = L_{cross}(y_w, \widehat{y}_w) \tag{23}$$

$$L_i = L_{cross}(y_i, \widehat{y}_i) \tag{24}$$

where $L_{cross}(\cdot)$ is cross-entropy loss function. Finally, the loss function of this model is as follows:

$$L = L_{topic} + L_w + L_i \tag{25}$$

IV. Experimental Setting

A. Dataset

We use public dataset Multi-WOZ 2.0 [34] to verify the effectiveness of our proposed model, which contains 10,438 human-human conversation sessions. The total corpus consists of 7 sub-topics of Attraction, Hospital, Police, Hotel, Restaurant, Taxi, and Train, and each dialogue contains 1 to 5 topics, and it includes 3,406 single-topic dialogues and 7,032 multi-topic dialogues with 24,071 unique tokens which fully shows the complexity of the corpus in order to train a complex generative model. Some statistics about the dataset are presented in Table 1.

TABLE I
DATASET STATISTICS

Description	Number
# Dialogues	8,438
Total # turns	115,424
Total # tokens	1,520,970
Avg. turns per dialogue	13.68
Avg. tokens per turn	13.18
Total unique tokens	24,071

B. Implementation Details

In experiments, the infused KG for our model contains 2,037 entities which consists of items and non-item words to represent the relationship between items and words. For each word, we use the name as its feature to feed into GCN network, while for each item, the attribute values of the item are fed into GCN network. For example, the item <Attraction> consists of attributes like <address>, <postcode>, <phone>, <area> and <type>. We implement this model in Pytorch and train on Tesla P100 16GB. And we train the model with a learning rate of 1e-2, and max epoch of 50. Mini-batch SGD with a batch size of 64 and Adam is used as the optimization algorithm with a weight decay of 1e-2. The dimension of graph or utterance embedding is set to 64. The layer size of GCN L is set to 2. The word embedding is initialized from pre-trained BERT embeddings and fine-tuned during training.

C. Metrics

We evaluate the model based on automatic evaluations and human evaluations as [35] in the conversation and the recommendation. For automatic evaluation, we use BLEU, perplexity and distinct n-gram to evaluate. BLEU is used to compare the alignment between the generated sentence and the truth response. Perplexity is a measurement for the fluency of natural language. Lower perplexity refers to higher fluency. Distinct n-gram is a measurement for the diversity of response. Specifically, we use Distinct 2-gram, 3-gram and 4-gram at the corpus level to evaluate the diversity. In addition, the evaluation metric of recommendation is Accuracy, which evaluates the accuracy of predicted items in generated system responses. For human evaluation, we use Fluency, Informativeness, Appropriateness, Proactivity in turn-level to evaluate. The Fluency measures if the generated sentence is smooth and coherent. The Informativeness measures if the model full use of knowledge in the response. The Appropriateness measures if the generated sentence is consistent with the topic of history dialogue. The proactivity measures if the model can introduce new topics with good fluency and coherence.

D. Baselines

Baseline models used in the experiment are briefly stated as follows.

(1)**HRED** [36]: It applies the hierarchical end-to-end architecture, with one layer modeling the word-level and the other layer modeling the utterance-level without attention mechanism.

(2)**Transformer** [18]: It applies a Transformer-based encoder-decoder model which can understand conversations and generate responses without information from the recommendation module.

(3)**TopicRNN** [27]: This model captures local dependency through RNNs and global semantic using latent topic representation.

(4)**ReDial** [37]: This model consists of a dialogue generation module based on HRED, a recommendation module and a sentiment analysis module. For recommendation module, it only considers relationship between items while ignoring non-item words in history dialogue.

(5)**DCR** [1]: It augments the seq2seq models with a neural latent topic component to guide response generation. For recommendation, it leverages GCNs to capture the relationship between items and get the representation of items. It is not considering non-item words in graph structure.

(6)**KBRD** [16]: It uses knowledge graph to enrich the user representation for recommendation. It bridges the gap between recommendation system and dialogue system via knowledge propagation.

V. Results

A. Evaluation

In this section, we evaluate the proposed KGTO in terms of recommendation accuracy, response generation quality and human evaluation.

1) Recommendation: To evaluate the effectiveness of our recommendation module, we conduct an evaluation of Item Accuracy, and present the results in Table 2. A comparative analysis based on Table 2 is given. First, our proposed KGTO is superior to all baseline methods, including text generation and CRS methods, in terms of item accuracy, which shows that KGTO can improve the accuracy of recommendation. Second, ReDial, DCR, KBRD and KGTO perform quite better than HRED, Transformer and TopicRNN, because these four models are able to learn the representation of items for recommendation. Third, similar to DCR, topic information is also used in our model, but our model uses hierarchical attention mechanism to learn latent topic and has a 20.66% (=(22.19-18.39)÷18.39×100%) improvement, which shows that hierarchical attention mechanism can pay much attention to the keyword information describing items and learn better topic information to guide recommendation. What's more, the fusion of co-occurrence graph and knowledge is helpful for learning the relationship between items and words and learning a high-quality item representation. Fourth, compared with KBRD, our model has a 5.47% improvement. This is because the topic of conversation is able to be captured by our model, and the topic is further used to guide GCNs to get the representation of items with topic weight information, and then selects items related to the topic for recommendation. Finally, we analyze the ablation experiment and find that the performance of the KGTO/HA is better than the KGTO/GraphFusion, which shows that the GraphFusion operation is helpful for learning a good representation of item. However, they are not both as effective as KGTO, indicating that the integration of hierarchical attention mechanism and GraphFusion is more effective for recommendation.

TABLE II
Performance of Recommendation

Model	Accuracy
HRED	2.61
Transformer	6.53
TopicRNN	4.56
ReDial	16.91
DCR	18.39
KBRD	21.04
KGTO	**22.19**
KGTO/HA[a]	21.95
KGTO/GraphFusion[b]	21.27

[a]KGTO without Hierarchical Attention module.
[b]KGTO without GraphFusion operation.

2) Dialogue Generation: The result of evaluation on dialogue generation summarizes in Table 3. By analyzing the Table 3, observations of the experiment are given. First, our proposed method outperforms all the baselines, which shows that our model can improve the quality of dialogue generation, including the consistency and the diversity of the dialogue. Second, our model is better than the traditional text generation methods, showing that learning the relationship and representation of items is needed for enhancing the informativeness of dialogue, to get a good system response. Third, our model achieves some improvements over all the basic CRS model. Compared with DCR, the performance of our model is much better than DCR in BLEU, Dist-2, Dist-3 and Dist-4, by 9.50%, 68.76%, 7.69% and 13.77% respectively. Because our model uses word-level attention mechanism to filter out words without content and information, such as stop words, to learn a better topic distribution information to guide the selection of the words or items. What's more, the fusion of item-word co-occurrence graph and item-item knowledge graph is important to enrich the contextual information of items and makes the word and item in the generated response more relevant. In addition, our model performs better than KBRD in BLEU, Dist-2, Dist-3 and Dist-4, by 4.90%, 12.43%, 4.31% and 3.01% respectively. Because our model can capture topic information to guide response generation while KBRD does not consider topic information, showing that the latent topic is helpful for global topic control and response generation. Last but not least, by analyzing ablation experiments, we find that the performance of KGTO/GraphFusion is better than the KGTO/HA in BLEU and Distinct, which indicates that the more accurate topic is more helpful than the GraphFusion operation for maintaining the consistency of the conversation and enhancing the diversity of dialogue. However, both of them are not able to achieve the performance of KGTO, showing that mutual promotion of two modules will produce a better system response.

3) Human Evaluation: Table 4 presents the result of human evaluation on response generation. As shown in Table 4, we can find that our model performs well compared to other

TABLE III
PERFORMANCE OF DIALOGUE GENERATION

Model	BLEU	Dist-2	Dist-3	Dist-4	PPL
HRED	10.62	2.23	6.37	22.53	19.32
Transformer	15.25	2.56	9.76	26.71	18.06
TopicRNN	16.39	2.24	10.98	27.92	13.01
ReDial	18.91	3.72	20.69	31.71	16.16
DCR	24.63	4.61	27.17	36.96	11.01
KBRD	25.71	6.92	28.05	40.82	10.32
KGTO	**26.97**	**7.78**	**29.26**	**42.05**	**9.77**
KGTO/HA	26.03	7.63	28.62	41.71	10.19
KGTO/GraphFusion	26.85	7.71	29.03	41.89	9.92

CRS models. In terms of fluency and appropriateness, our model performs better than DCR and KBRD, this is because our model can filter out words without information through hierarchical attention mechanism to capture the global topic information. In terms of informativeness, our model has 20.6% and 6.3% improvement over DCR and KBRD. This is due to the fusion of contextual information in items, which fully explores the relationship between item and word, and the generated response will not be templated, such as "Yes, it is.", "I do not know." and so on. In addition, the fusion of the co-occurrence graph with knowledge graph can associate items with keywords, and these words can be used as the explanation information of the recommended item to be generated in the response, which aggravates more explanatory information in the response. In terms of proactiveness, the topic can not be captured well in KBRD, so the proactiveness is insufficient, but our model can capture the change of the topic information in time, so as to guide the generation of responses.

TABLE IV
PERFORMANCE OF DIALOGUE GENERATION

Model	Fluen.	Informa.	Appropri.	Proacti.
ReDial	1.21	0.49	0.16	0.57
DCR	1.63	1.26	0.33	0.92
KBRD	1.78	1.43	0.46	1.16
KGTO	**1.86**	**1.52**	**0.59**	**1.21**

B. Model Interpretability

We present a case to show how our model works. As shown in Figure 3, we calculate the weight of each word to each sentence (blue number in Figure 3) to get the contribution of each word to each sentence, and then select keywords based on the learned threshold to learn the topic distribution of history dialogue. In this example, the predicted topic is finding hotels (green histogram in Figure 3), and we use different models to decode history dialogue to get the response. And our observations based on Figure 3 are as follows.

In general, for the text generation methods like HRED, Transformer and TopicRNN, we find that the generated responses do not contain the item because it does not contain the recommendation module, and for the CRS methods, the model contains recommended items and dialogue information compared with text generation methods. Then, our model is more informative than DCR, because our model fuses co-occurrence graph and knowledge graph, which can make each item associated with some words, and these words are regarded as the explainable information of item in generated responses. And our model can generate much more topic words than the KBRD model, because our model filters out words without information to capture more accurate topic information, to further guide the selection of words or items. Finally, the consistency and diversity of our model are far from those of human-like language, and we still need to work hard on it.

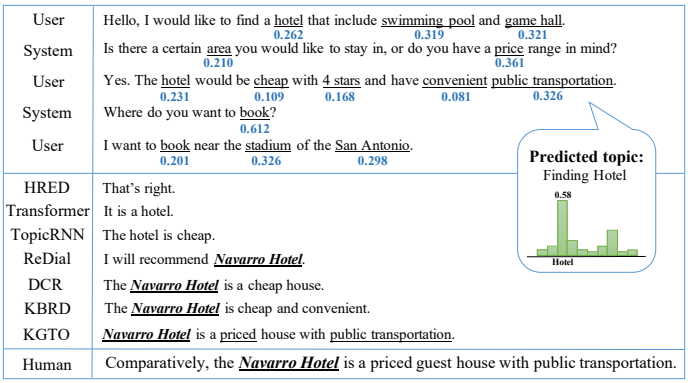

Fig. 3. An example of explainable model.

VI. CONCLUSION

We proposed a model named KGTO, a CRS model that can capture latent topic information to assistant the selection of items and words in generating responses. In addition, in order to solve the problem of insufficient contextual information in items, we fuse the co-occurrence graph and the knowledge graph to explore the relationship between items and words, and learn the representation of items with contextual information through graph convolutional networks. Finally, we incorporate information from the dialogue and the recommendation into the response generation module to generate responses that are consistent with the topic. The results of experiences show the effectiveness of our model.

REFERENCES

[1] L. Liao, R. Takanobu, Y. Ma, X. Yang, M. Huang, and T.-S. Chua, "Deep conversational recommender in travel," *arXiv preprint arXiv:1907.00710*, 2019.

[2] W. Ma, R. Takanobu, M. Tu, and M. Huang, "Bridging the gap between conversational reasoning and interactive recommendation," *arXiv preprint arXiv:2010.10333*, 2020.

[3] K. Zhou, W. X. Zhao, S. Bian, Y. Zhou, J.-R. Wen, and J. Yu, "Improving conversational recommender systems via knowledge graph based semantic fusion," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1006–1014.

[4] K. Christakopoulou, A. Beutel, R. Li, S. Jain, and E. H. Chi, "Q&r: A two-stage approach toward interactive recommendation," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 139–148.

[5] X. Zeng, J. Li, L. Wang, Z. Mao, and K.-F. Wong, "Dynamic online conversation recommendation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 3331–3341.

[6] D. Jannach, A. Manzoor, W. Cai, and L. Chen, "A survey on conversational recommender systems," *arXiv preprint arXiv:2004.00646*, 2020.

[7] W. Lei, X. He, M. de Rijke, and T.-S. Chua, "Conversational recommendation: Formulation, methods, and evaluation," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 2425–2428.

[8] C. Gao, W. Lei, X. He, M. de Rijke, and T.-S. Chua, "Advances and challenges in conversational recommender systems: A survey," *arXiv preprint arXiv:2101.09459*, 2021.

[9] Y. Sun and Y. Zhang, "Conversational recommender system," in *The 41st international acm sigir conference on research & development in information retrieval*, 2018, pp. 235–244.

[10] Y. Zhang, X. Chen, Q. Ai, L. Yang, and W. B. Croft, "Towards conversational search and recommendation: System ask, user respond," in *Proceedings of the 27th acm international conference on information and knowledge management*, 2018, pp. 177–186.

[11] W. Lei, X. He, Y. Miao, Q. Wu, R. Hong, M.-Y. Kan, and T.-S. Chua, "Estimation-action-reflection: Towards deep interaction between conversational and recommender systems," in *Proceedings of the 13th International Conference on Web Search and Data Mining*, 2020, pp. 304–312.

[12] R. S. Sutton, A. G. Barto *et al.*, *Introduction to reinforcement learning*. MIT press Cambridge, 1998, vol. 135.

[13] Q. Y. E. Lim, Q. Cao, and C. Quek, "Dynamic portfolio rebalancing through reinforcement learning," *Neural Computing and Applications*, pp. 1–15, 2021.

[14] D. Vandic, S. Aanen, F. Frasincar, and U. Kaymak, "Dynamic facet ordering for faceted product search engines," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 5, pp. 1004–1016, 2017.

[15] S. Rendle, "Factorization machines," in *2010 IEEE International conference on data mining*. IEEE, 2010, pp. 995–1000.

[16] Q. Chen, J. Lin, Y. Zhang, M. Ding, Y. Cen, H. Yang, and J. Tang, "Towards knowledge-based recommender dialog system," *arXiv preprint arXiv:1908.05391*, 2019.

[17] Z. Chen, X. Wang, X. Xie, M. Parsana, A. Soni, X. Ao, and E. Chen, "Towards explainable conversational recommendation," in *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2021, pp. 2994–3000.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[19] K. Zhou, Y. Zhou, W. X. Zhao, X. Wang, and J.-R. Wen, "Towards topic-guided conversational recommender system," *arXiv preprint arXiv:2010.04125*, 2020.

[20] F.-Y. Sun, J. Hoffmann, V. Verma, and J. Tang, "Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization," *arXiv preprint arXiv:1908.01000*, 2019.

[21] T. Zhang, Y. Liu, P. Zhong, C. Zhang, H. Wang, and C. Miao, "Kecrs: Towards knowledge-enriched conversational recommendation system," *arXiv preprint arXiv:2105.08261*, 2021.

[22] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2016, pp. 1480–1489.

[23] J. Deng, L. Cheng, and Z. Wang, "Self-attention-based bigru and capsule network for named entity recognition," *arXiv preprint arXiv:2002.00735*, 2020.

[24] S. Jayalakshmy and G. F. Sudha, "Gtcc-based bilstm deep-learning framework for respiratory sound classification using empirical mode decomposition," *Neural Computing and Applications*, vol. 33, no. 24, pp. 17 029–17 040, 2021.

[25] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[26] H. Xu, S. Moon, H. Liu, B. Liu, P. Shah, and P. S. Yu, "User memory reasoning for conversational recommendation," *arXiv preprint arXiv:2006.00184*, 2020.

[27] A. B. Dieng, C. Wang, J. Gao, and J. Paisley, "Topicrnn: A recurrent neural network with long-range semantic dependency," *arXiv preprint arXiv:1611.01702*, 2016.

[28] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[29] F. Huang, S. Zhang, J. Zhang, and G. Yu, "Multimodal learning for topic sentiment analysis in microblogging," *Neurocomputing*, vol. 253, pp. 144–153, 2017.

[30] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang, and S. Qiao, "Attention-emotion-enhanced convolutional lstm for sentiment analysis," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[31] Y. Miao, L. Yu, and P. Blunsom, "Neural variational inference for text processing," in *International conference on machine learning*. PMLR, 2016, pp. 1727–1736.

[32] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[33] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[34] P. Budzianowski, T.-H. Wen, B.-H. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gašić, "Multiwoz–a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling," *arXiv preprint arXiv:1810.00278*, 2018.

[35] Z. Liu, H. Wang, Z.-Y. Niu, H. Wu, W. Che, and T. Liu, "Towards conversational recommendation over multi-type dialogs," *arXiv preprint arXiv:2005.03954*, 2020.

[36] I. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[37] R. Li, S. Kahou, H. Schulz, V. Michalski, L. Charlin, and C. Pal, "Towards deep conversational recommendations," *arXiv preprint arXiv:1812.07617*, 2018.