


ARTICLE

Improved conversational recommender system based on dialog context

Xiaoyi Wang^{1,2} , Jie Liu^{2,3} and Jianyong Duan³

¹School of Literature, Capital Normal University, Beijing, China, ²China Language Intelligence Research Center, Beijing, China, and ³School of Information Science, North China University of Technology, Beijing, China

Corresponding author: Jie Liu; Email: liujxxy@126.com

(Received 13 August 2022; revised 9 August 2023; accepted 10 August 2023; first published online 8 September 2023)

Abstract

Conversational recommender system (CRS) needs to be seamlessly integrated between the two modules of recommendation and dialog, aiming to recommend high-quality items to users through multiple rounds of interactive dialogs. Items can typically refer to goods, movies, news, etc. Through this form of interactive dialog, users can express their preferences in real time, and the system can fully understand the user's thoughts and recommend corresponding items. Although mainstream dialog recommendation systems have improved the performance to some extent, there are still some key issues, such as insufficient consideration of the entity's order in the dialog, the different contributions of items in the dialog history, and the low diversity of generated responses. To address these shortcomings, we propose an improved dialog context model based on time-series features. Firstly, we augment the semantic representation of words and items using two external knowledge graphs and align the semantic space using mutual information maximization techniques. Secondly, we add a retrieval model to the dialog recommendation system to provide auxiliary information for generating replies. We then utilize a deep timing network to serialize the dialog content and more accurately learn the feature relationship between users and items for recommendation. In this paper, the dialog recommendation system is divided into two components, and different evaluation indicators are used to evaluate the performance of the dialog component and the recommendation component. Experimental results on widely used benchmarks show that the proposed method is effective.

Keywords: Retrieval and generation; Dialog recommender system; Time-series features

1. Introduction

With the popularity of the Internet and information technology, the network information data show an explosive growth and information overload becomes prominent. As users are inundated with a large and growing pool of content, products, and services (collectively referred to as items), recommender systems have become essential tools for selecting the information they need (Ricci, Rokach, and Shapira 2021). These recommender systems are a static process that tend to learn users' interests from their past behaviors, thereby recommending suitable items (Chen *et al.* 2019). However, the user's current interest may have changed, and it is difficult to provide feedback on dissatisfaction with current recommended content or new requirements in the current interaction. The dialog system is more concerned about the user's current interests (Christakopoulou,

Xiaoyi Wang and Jie Liu contributed equally to this work.

Table 1. Two examples of dialog recommender systems for movies

Role	Example 1	Example 2
User	I am looking for a movie	I am looking for a movie
Recommender	The <i>Sixth Sense</i> is quite popular. Maybe you will like it	Which actor do you like?
User	This movie sounds scary, so I hate it	I like a lot, such as Bruce Willis and Henry Thomas
Recommender	Why don't you look at <i>Pulp Fiction</i> , a funny comedy	The <i>Sixth Sense</i> might be good for you, a horror movie
User	That sounds good, which actor starred it?	This movie sounds scary
Recommender	Both of these are Bruce Willis	It was played by Bruce Willis . Try it out
User	Great! I like Bruce Willis , and I will look at both. Thank you!	Great! Thank you!

Radlinski, and Hofmann 2016). In the process of user decision-making, both historical information and current information are significant, so the dialog recommender system emerges as the times require.

Conversational recommender system aims to recommend high-quality items to users through interactive dialog, which realizes the linkage between traditional information retrieval and recommendation (Jannach *et al.* 2021). Specifically, the system has multiple rounds of real-time interaction based on natural language with the user, combining the user's historical interests with their preferences captured in the current conversation to recommend products. Knowledge graph contains item attributes and various types of relationships, providing powerful background knowledge that can provide rich semantic information to recommendation algorithms. Therefore, dialog recommendation systems based on knowledge graphs have tremendous potential (Li *et al.* 2018).

While existing research has made significant strides in improving the performance of conversational recommender systems, certain challenges remain. In particular, existing systems often overlook the importance of time-series feature information in capturing user interests and recommending relevant items, where time-series features refer to the change rule or trend of user preferences over time. Additionally, systems often fail to consider the content of the dialog when making recommendations, leading to the loss of overall semantic meaning. For example, different mentions of the same item may have positive or negative connotations, leading to varying recommendations. The items are mentioned in different order, the suggestions may also be different, so we should take information about the content of the conversation into account in the recommendation. As shown in Table 1, both examples mentioned that the user likes Bruce Willis, but due to the time sequence mentioned is inconsistent, the final system recommendation results are also different. At the same time, we also pay attention to the connection between words (such as movies and actors) and introduce knowledge graph (Wang *et al.* 2019) and copy mechanism (He *et al.* 2017a) to enrich dialog information.

The use of deep learning methods to study natural language processing has sprung up in recent years. We use deep learning methods to propose a deep timing network (DTN) to model the serialized dialog data, capture global information and current interests, and use the copy mechanism to map some proper nouns into the generated sequence, so as to give appropriate recommendations. We demonstrate through a series of experiments that the proposed model achieves certain improvements in both recommender and conversation tasks.

The main contributions of this paper are as follows:

- We focus on the phenomenon of user interest transition in dialog recommender systems and propose a new network structure to optimize user interest recommendation and improve accuracy.
- We propose to add a retrieval model and use the copy mechanism to retain retrieved information in the output, enhancing the generation model's effectiveness and alleviating the problem of new words.
- We propose DTN, which uses BiGRU attention to serialize dialog content, accurately learns the feature relationship between users and items, and adds position encoding to absorb position information, reinforcing the importance of time-series features in the dialog context and reducing the occurrence of repeated words and increasing diversity in the generated responses.

2. Related work

The conversation recommender system aims to facilitate task-oriented multi-turn dialogs with users, consisting of two modules: recommendation and dialog (Lei *et al.* 2020). In this section, we provide a detailed introduction to these two components, as well as discuss related work in the field of dialog recommender systems.

2.1 Recommender system

The recommendation system mainly utilizes user's behavioral information on items to mine personalized needs and actively provides them with information that satisfies their needs through their interest model (Christakopoulou *et al.* 2018). Traditional personalized recommendation algorithms are mainly divided into content-based recommendation algorithms and collaborative filtering recommendation algorithms. Content-based filtering (Javed *et al.* 2021) requires analyzing the description of file resources and each user's interests, thus establishing a user preference model and providing recommendation services to users through their interest preference model. The core idea of recommender system based on collaborative filtering (Polatidis and Georgiadis 2016) is to integrate the explicit feedback information of users and items, and filter out the items that may be of interest to target users for recommendation.

With the development of deep learning technology, He *et al.* (2017b) proposed a neural collaborative filtering model, integrating matrix decomposition processing with deep learning. Liu *et al.* (2020) used CNN to extract image information, learn the impact of product images on user behavior, and improve the accuracy of click probability prediction. Feng *et al.* (2019) applied RNN to capture dynamic and constantly changing user interests from user behavior sequences. Zhang *et al.* (2021) adopted the attention mechanism to alleviate the uninterpretability of the recommendation model and improve the sense of user experience. However, it is important to note that the number of users and items in commercial recommendation systems is often very large, and users can only access a very limited number of items, resulting in very sparse behavioral information about items. To address this problem, we introduce knowledge graph (Bizer *et al.* 2009) as auxiliary information to improve the performance of the recommendation system, which contains item attributes and relationships of various types, allowing us to obtain multi-dimensional, richer information about items and their relationships.

2.2 Conversational system

The field of dialog systems has received increasing attention in various domains and can be broadly classified into two types: task-oriented and non-task-oriented dialog systems

(Chen *et al.* 2017). Task-oriented systems are designed to help users complete practical and specific tasks and can be divided into retrieval-based and generation-based methods. Retrieval-based methods involve searching a pre-defined index and learning to select a response from the current dialog (Cai *et al.* 2019). Lowe *et al.* (2015) concatenated the dialog history with the question and encoded them separately with RNNs before calculating the match between the two encoded vectors to rank candidate responses. Kadlec, Schmid, and Kleindienst (2015) applied CNN and bidirectional LSTMs to the candidate response selection task and investigated the impact of semantic representation on reply selection. Yang *et al.* (2018) guided the selection of responses by retrieving the responses corresponding to the most similar questions of the user as auxiliary information. One potential advantage of retrieval-based methods is that they return utterances previously uttered by humans, which means that they are typically grammatically correct and inherently meaningful in terms of semantics. In addition, retrieval-based methods do not require the potentially expensive training of complex language models. On the other hand, retrieval-based methods may lack creativity and may suffer from the problem of not being able to respond appropriately to previously unseen situations.

Unlike retrieval-based dialog systems, generative methods can produce a completely new response (Singla *et al.* 2020) which is relatively more flexible. Vinyals and Le (2015) first used the Seq2Seq model in the dialog generation task, constructing the model by using two RNNs, one for encoding the user's message and the other for decoding the generated response. Shang, Lu, and Li (2015) adopted a general encoder-decoder framework that generates a response based on a potential representation of the input text as the decoding process, while using RNN for both encoding and decoding. Various extensions based on the Seq2Seq architecture have been developed to improve the quality of response generation. To overcome the limitations of data scale, Naous, Hokayem, and Hajj (2020) proposed an empathy-driven Arabic chatbot with a special encoder-decoder composed of a LSTM Sequence2Sequence. Boussakssou, Ezzikouri, and Erritali (2022) presented an Arabic chatbot focused on the Seq2Seq framework, using Aravec to transform a sentence into a vector of actual numbers representing the sentence's input. Woungang *et al.* (2023) established a generative model neural conversation system using a deep LSTM Seq2Seq model with attention mechanism. However, generative methods also have disadvantages, such as the tendency to produce grammatical errors or meaningless responses, so we propose combining both retrieval and generative methods.

2.3 Conversational recommender system

Dialog technology provides new solutions and methods for solving the problems existing in traditional recommendation systems, namely the conversational recommender system. Through rich interactive behavior, dialog recommendation breaks down the barriers of information asymmetry between the system and the user in static recommendation systems, allowing the recommender system to dynamically capture user preferences in interactive dialogs. The dialog recommendation system can also be called a task-oriented dialog system, in which the system can elicit detailed current preferences from the user, provide explanations for topic recommendations, or handle user feedback on proposed suggestions to achieve dynamic updates and learning.

Recently, some research has begun to reintegrate dialog and recommendation systems to recommend the appropriate items through dialog. Quadrana *et al.* (2017) proposed a hierarchical RNN model structure to explore relationships within and between dialogs, achieving more reliable next-item recommendations. Wang *et al.* (2020) introduced GNNs to model complex transitions within and between dialogs, building better-performing dialog recommender systems. Yuan *et al.* (2019) proposed the NextInet model based on generative models to design a probability distribution for candidate items. In particular, Li *et al.* (2018) collected a dialog dataset focused on providing movie recommendations, and Chen *et al.* (2019) proposed the KBRD model for end-to-end training of both recommendation and dialog systems, making it possible to suggest

recommendations based on entities mentioned in the dialog. Zhou *et al.* (2020) proposed KGSE, which used knowledge graphs and semantic fusion to integrate dialog and recommendation modules. Ma, Takano, and Huang (2021) proposed a model called CR-walk that performs tree-based reasoning on knowledge graphs and generates dialog actions to guide language generation.

Although these studies have improved the performance of dialog recommender systems, some limitations still exist, such as the failure to fully consider the order of entities mentioned in dialogs, and the current dialog recommender systems relying solely on generation methods, which require further improvement in terms of accuracy. Therefore, inspired by existing research on dialog recommender systems, we propose utilizing knowledge graphs to introduce external knowledge, using time-series feature methods to fully leverage contextual information, and incorporating retrieval methods (Manzoor and Jannach 2022) to assist in generating methods to jointly construct a dialog module. This will enable us to achieve a more reliable recommendation dialog text generation model and provide the most reasonable recommendations.

3. Task definition and approach

In this section, we start from the task definition and describe our approach in detail. The overall model framework is shown in Fig. 1, which is divided into three modules a, b, and c, corresponding to semantic fusion module, dialog module, and recommender module, respectively. In part a, the semantic representation is enhanced by encoding an external knowledge graph. Part b includes retrieval and generation components. In part c, the enhanced semantic representation is passed through DTN to obtain more accurate user preferences.

3.1 Task definition

The conversational recommender system, as the name implies, has to solve both dialog and recommendation problems. The overall architecture of a dialog recommender system can be thought of as consisting of a recommendation module and a dialog module. The goal of the entire system is to provide the most accurate recommendation results in the shortest possible number of dialogs.

Let $u \in U$ denote a user u in user set U , $i \in I$ denote an item i in item set I , and $v \in V$ denote a word v in vocabulary V . A dialog C is an ordered list $C = \{S_t\}_{t=1}^n$ composed of a series of sentences, and S_t is the sentence at the t -th round of dialog interaction. In the t -th round of the dialog, the recommendation module will select a candidate item subset I_t from the item set I according to a certain strategy, and the dialog module needs to generate the reply text S_t of the current round. It is worth noting that the candidate item subset I_t may be empty, in which case the dialog module generates relevant texts for queries, or just chat texts. Given the n sentences that the above system interacts with the user, the goal of the generative dialog recommender system is to generate corresponding responses. That is, the recommendation result I_{n+1} and the reply text S_{n+1} together constitute the reply at this moment.

3.2 Semantic fusion module

The semantic fusion module is constructed based on knowledge graph(KG) and graph convolutional neural network(GCN). To comprehensively understand user interests, two independent knowledge graphs are used in dialog and recommender systems to enhance the representation of basic semantic units. ConceptNet (Speer, Chin, and Havasi 2017) is a common sense knowledge base that represents the most basic knowledge that humans understand. It is composed of relational knowledge in the form of triples and focuses on the relationship between words, which is used as a word-oriented KG. Conversation-related words are filtered from the entire KG to form a small KG, which provides relationships between words, such as synonyms,

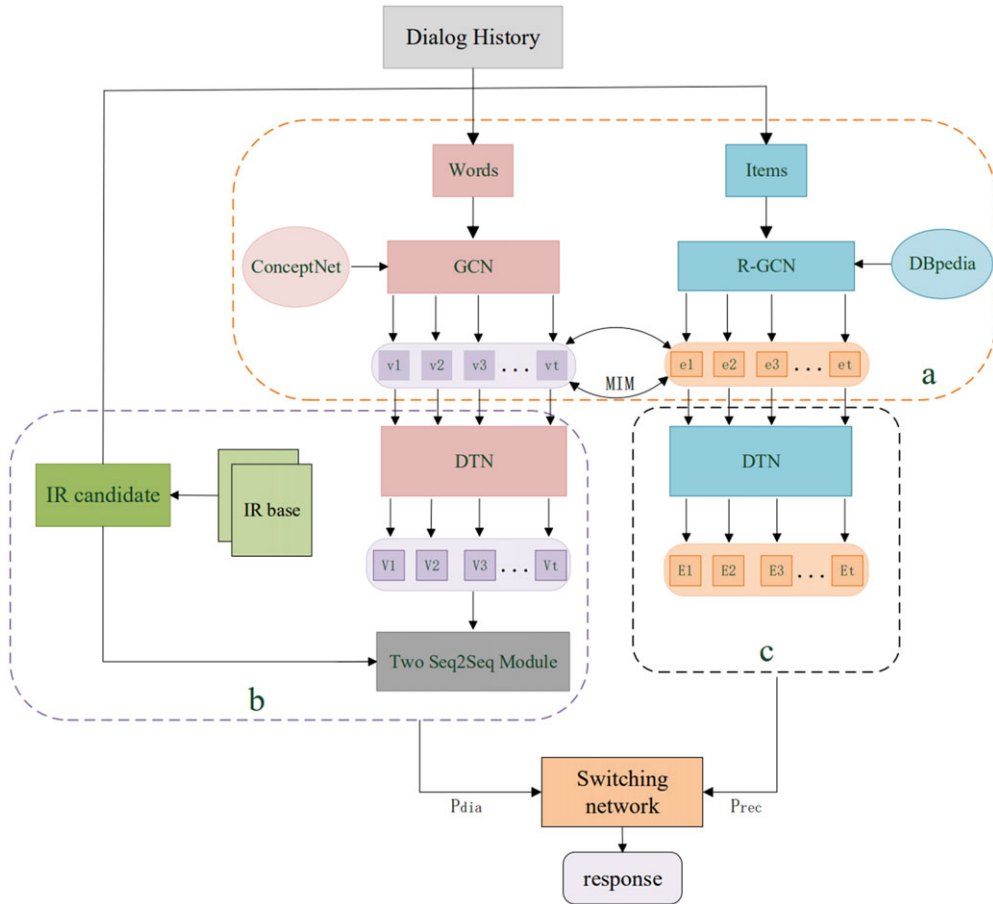


Figure 1. Overall framework of the model.

antonyms, and co-occurring words for each word. GCN (Chiang *et al.* 2019) is then employed to learn embedding representations for word nodes. In the data of the graph structure, both the characteristics of each node and structural information should be fully considered. GCN can automatically learn the feature information and structural information of the graph, and each node contains its own characteristics and structural information. On each update of node representation, GCN receives information from one-hop neighbors in the graph and performs the following aggregation operations:

$$V^{(l)} = \text{ReLU} \left(D^{-\frac{1}{2}} A D^{-\frac{1}{2}} V^{(l-1)} W^l \right) \tag{1}$$

where $V^{(l)} \in \mathbb{R}^{V \times d_w}$ is the representation of nodes, W^l is the learnable matrix of each layer, A is the adjacency matrix corresponding to the graph, and D is a diagonal matrix. By stacking multiple convolutions, information can be propagated together along the graph structure. When the algorithm terminates, each word corresponds to a d_w -dimensional representation n_w . Thus, this paper learns the embedding representation of word nodes through GCN.

On the other hand, we adopt DBpedia (Bizer *et al.* 2009) as an item-oriented knowledge graph that provides structured facts about item attributes and the relationships between them. Items related to the dialog content are filtered out from the entire KG to form a small KG, and R-GCN (Gao *et al.* 2020) is used to extract item representations on subgraphs that more fully consider

```
'messages': 'Hello|Good Morning|I'm looking for movies about the Royal Family @130573|Have
'conversationId': '16622',
'text': 'Thanks for the information Will check it out ',
'text_mentioned_movie_id': '',
'mentioned_movie_id': '@130573|@177724|@171529|@177605|@171529|@104192',
```

Figure 2. Data form diagram.

edge types and orientations. R-GCN is a simple attempt of GCN in multi-relational graph scenarios. From homogeneous graphs to heterogeneous graphs, R-GCN solves the core problem of interacting with multiple relations. Under each relationship, the inward-pointing and outward-pointing points are regarded as neighbor points, and self-circulating features are added at the same time to perform feature fusion and participate in updating the central node. They are beneficial for handling multi-relational data features in knowledge bases, so R-GCN is used to extract item representations on subgraphs. After obtaining the node representations of word and item, in order to effectively align the semantic space of the two KGs, we use mutual information maximization (Yeh and Chen 2019) to mutually enhance the data representation of paired signals and make the representations of words and items appearing in a dialog more similar, thereby retaining the most important features.

3.3 Dialog retrieval module

Retrieval-based methods are commonly used to select the desired response from a candidate response pool using a dialog corpus and user questions. In this paper, we construct a retrieval module and create a new retrieval database that includes the dialog history in the training set, the ID and content of each sentence, and the movie ID mentioned in each sentence and the dialog history. To avoid problems with repeated lists, maybe this sentence refers to a movie, and the next sentence also refers to it, we use Elasticsearch (Divya and Goyal 2013), an open-source search engine based on the Lucene library, which processes data in JSON format. User data as shown in Fig. 2 below:

Our retrieval module retrieves a set of candidate responses from the dialog history repository using an entity-context matching method. We use Elasticsearch to index all conversation histories that contain the entity mentioned in the current conversation in the training data, which improves search efficiency by finding the target document's approximate position directly in memory. We then calculate the similarity between the searched question and the input question to obtain the most similar questions' replies as candidate responses. In summary, our retrieval module feeds the dialog content in the training set into a third-party retrieval interface, extracting informative words from the retrieved responses to generate the most useful response.

3.4 Dialog generation module

To leverage the temporal nature of dialog content, we propose the deep timing network (DTN). As depicted in Fig. 3, the word and item vectors are fed into the position encoding module, followed by feature extraction through BiGRU to obtain the feature vectors of the word and item. Then, the attention mechanism assigns corresponding probability weights to different feature vectors, and the keyword and key item information are obtained through softmax calculation. The representation after fusion of the knowledge graph is then passed through the DTN, which endows the dialog data with location information and time-series features. The attention mechanism is used to capture global information and current interests to infer user preferences. Moreover, we

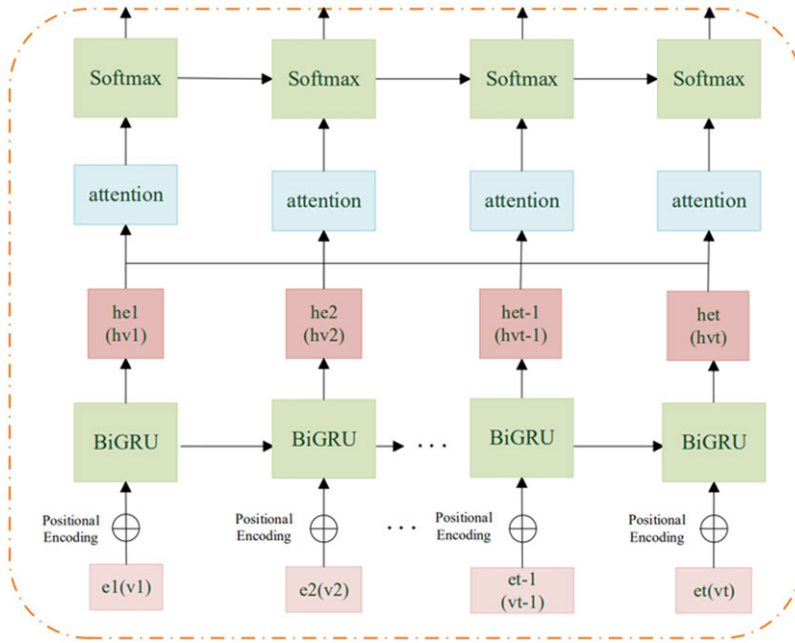


Figure 3. Deep Timing Network(DTN).

suggest introducing relevant information retrieved by our retrieval module to provide appropriate recommendations for the generation module.

3.4.1 Positional encoding

The content of a dialog is a time-series data, and the order and position of words in a sentence hold significant importance. These factors are not only integral to the grammatical structure of a sentence but also crucial for expressing its semantics. In fact, the same set of words in different positions within the same sentence can convey vastly different meanings. For example, in the following two sentences, the constituent words of the sentences are exactly the same, but the meanings are quite different:

I don't like the story of this movie, but I like the actors.

I like the story of this movie, but I don't like the actors.

To prevent unnecessary misunderstandings, we introduce position encoding (Yang *et al.* 2016) when modeling text data to encode the positions of words in the sequence. This allows the model to fully comprehend the relative relationship between positions. In this paper, we express the sine and cosine functions of different frequencies as follows:

$$PE(pos, 2i) = \sin \left(pos/10,000^{2i/d_{model}} \right) \tag{2a}$$

$$PE(pos, 2i + 1) = \cos \left(pos/10,000^{2i/d_{model}} \right) \tag{2b}$$

where PE is a two-dimensional matrix, with the same size as the input embedding dimension. The rows represent the words, and the columns represent the word vectors. pos represents the position of the word in the sentence; d_{model} represents the dimension of the word vector; i represents the position of the word vector. The above formula means adding the sin variable to the even-numbered position of the word vector of each word, and adding the cos variable to the

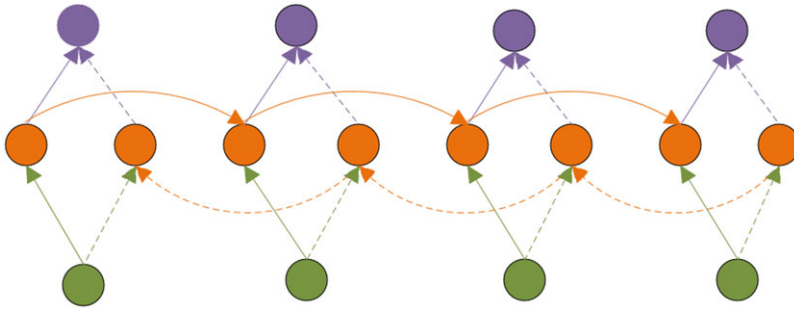


Figure 4. BiGRU structure diagram (Bansal *et al.* 2016).

odd-numbered position, filling the entire PE matrix, and then adding it to the input embedding. This introduces position information of words, reducing the occurrence of repeated words in the generated replies.

3.4.2 BiGRU-attention

GRU, a type of RNN, determines the output value of the current node jointly based on the current input and the output of the previous node. This enables it to fully capture the information between the dialog content before and after (Bansal, Belanger, and McCallum 2016). However, GRU can only use historical information to make judgments on the current information and cannot use future information. Therefore, after assigning relative position information to the word and item-level data representations, we use bidirectional GRU to deeply extract the input vector (as shown in Fig. 4), such that the output of the current moment is related to the state of the previous and next moments.

In addition, the attention mechanism (Vaswani *et al.* 2017) is introduced to assign corresponding probability weights to different word vectors and further extract features that highlight the key information of the dialog content. The attention mechanism helps to allocate resources to relatively more important tasks and alleviate information overload when computation is limited. It can effectively deal with the problem of complicated information selection and improve the ability of task processing. In each step of GRU, the attention mechanism can enhance the influence of relative interest, and the calculation of items in the recommender system is similar. Since the BiGRU model is regarded as two GRUs in opposite directions, the formula is simplified to Equation (3a). The word vector of the t -th word of the j -th sentence input at the i -th moment in the dialog system is v_{ijt} , and the specific calculation formula is:

$$h_{ijt} = \text{BiGRU}(v_{ijt}) \quad (3a)$$

$$u_{ijt} = \tanh(w_w h_{ijt} + b_w) \quad (3b)$$

$$\alpha_{ijt} = \frac{\exp(u_{ijt}^T u_w)}{\sum_t \exp(u_{ijt}^T u_w)} \quad (3c)$$

$$c_{ijt} = \sum_{i=1}^n \alpha_{ijt} h_{ijt} \quad (3d)$$

where h_{ijt} is the output vector of the BiGRU layer; w_w is the weight coefficient; b_w is the bias coefficient; u_w is the randomly initialized attention mechanism matrix, and the activation function

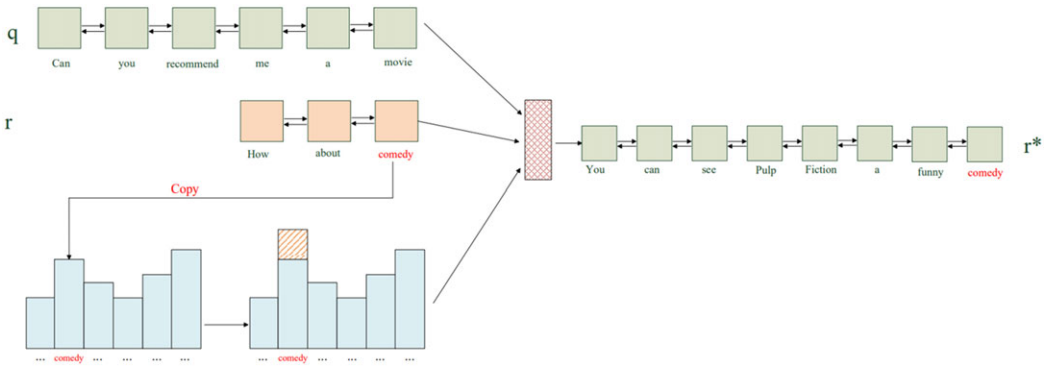


Figure 5. The two-seq2seq model.

tanh adjusts the neural network’s output, compressing the value between -1 and 1. The attention mechanism matrix is obtained by the cumulative sum of the products of different probability weights assigned by the attention mechanism and the states of each hidden layer, using the softmax function for normalization. Thus, we use the attention mechanism to capture global information and current interests to infer user preferences.

3.4.3 Copy mechanism

Traditional sequence-to-sequence (seq2seq) models often generate responses that are irrelevant or invalid due to the fact that they rely solely on the conditional probability of the given query. To address this issue, we propose a two-seq2seq model (shown in Fig. 5) that utilizes both the query q and the retrieved reply r to synthesize a customized reply r^* . Our proposed approach complements retrieval-based methods and enables generation-based dialog systems to generate new and relevant sentences. The two-seq2seq model consists of two encoders, one for the query and one for the retrieved reply, and a decoder that takes the outputs of both encoders as input. This design enables retrieval-based replies to provide additional information for generating responses, reducing the likelihood of generating generic replies. To fully capitalize on the retrieved replies, we integrate a copy mechanism (He *et al.* 2017a) into the decoding process.

The copy mechanism enhances the quality of the generated response r^* by extracting informative words from the retrieved reply r and suitable words from the encoder, which are then used as output words in the decoding process. The probability in the decoding process consists of two parts: the original probability P_g , and P_{r^*} , which represents the matching degree between the current state vector y_t and the corresponding state of the encoder. The formula is as follows:

$$P(y_t | s_t) = P_g(y_t | s_t) + P_{r^*}(y_t | h_{y_t}) \tag{4a}$$

$$P_{r^*}(y_t | h_{y_t}) = \delta(s_t W_c h_{y_t}) \tag{4b}$$

where h_{y_t} is the hidden state of the retrieved reply r , corresponding to y_t in the decoder, $\delta(\cdot)$ is the sigmoid function, and w_c is the parameter matching s_t and y_t . It should be noted that only words are copied from the replies, if y_t does not appear in the retrieved replies, then the corresponding P_{r^*} is 0. It can be seen that the generated responses are closely related to the query, and keywords are extracted from the retrieved responses, enhancing the quality of the generated responses.

In summary, our proposed approach combines DTN with a two-seq2seq model to enhance the relevance and coherence of the generated responses in a dialog system. By incorporating temporal features and leveraging the information contained in retrieved replies, we aim to improve the overall quality of the system’s outputs.

3.5 Enhanced recommender system

After obtaining the representation of the time-series feature of the conversation, our model generates recommendations based on the user's preferences, keyword information, and relevant item information. Specifically, we fuse the word representation $v^{(C)}$ of all contexts in the dialog with the representation $n^{(C)}$ of all items that appear and use this information to obtain the user preference p_u as follows:

$$p_u = \beta \cdot v^{(C)} + (1 - \beta) \cdot n^{(C)} \quad (5a)$$

$$\beta = \sigma \left(W_{\text{gate}} \left[v^{(C)}; n^{(C)} \right] \right) \quad (5b)$$

where β is the update gate, which is used to decide whether to ignore the current word, σ means that the update gate adds the two parts of information and puts it into the sigmoid activation function, and W_{gate} is a learnable parameter matrix. Once the user preference p_u has been obtained, we can calculate the probability of recommending each item to the user, resulting in a ranking of recommended items:

$$P_{\text{rec}}(i) = \text{softmax} \left(p_u^T \cdot n_i \right) \quad (5c)$$

With this framework, we have successfully integrated the recommender system and dialog system, promoting seamless collaboration between the two modules and improving the overall performance of the dialog recommender system.

4. Experiment

In this section, we provide a comprehensive description of our experimental setup, including datasets, results, and analysis.

4.1 Datasets

In order to verify the effectiveness of the model, we select REcommendations through DIALog (REDIAL) (Li *et al.* 2018) and INSPIRED (Hayati *et al.* 2020) as datasets. The REDIAL has 10,006 dialogs and 182,150 sentences on the topic of providing movie recommendations. The total number of users and movies is 956 and 51,699, respectively. In each dialog, one person is the movie seeker and the other is the recommender. The movie seeker must explain what genres they like and ask for advice on movies, while the recommender tries to understand the seeker's movie tastes and recommend movies. All information exchanges and suggestions are done using formal natural language, only talking about the movie and especially not the task itself. The training set, validation set, and test set are divided in a ratio of 8:1:1. INSPIRED is a similar dataset for movie recommendations as REDIAL, but smaller, containing only 1001 human conversations. In addition to the movie datasets, we also introduce related entities and relations from Conceptnet and DBpedia to improve the performance of our model. We systematically proposed and evaluated neural models for the entire dialog recommender system using this corpus.

4.2 Setting

This experiment adopts the PyTorch deep learning framework and uses the Python language programming to realize; the experimental running environment is JetBrains PyCharm software, ubuntu20.04 system, memory 11 GB, etc. The embedding dimensions (including hidden vectors) of the dialog system and the recommender system are set to 300 and 128, respectively; the number of layers L of both GCN and R-GCN is 1. During training, we use the Adam optimizer with default

Table 2. Experimental hyperparameter settings

Learning rate	0.001
Batch size	32
Position embedding size	50
GCN layer	1
R-GCN layer	1
gradient	[0,0.1]

parameter settings: the batch size is set to 32, the learning rate is 0.001, and the gradient is limited to [0, 0.1]. The experimental parameter Settings are shown in Table 2.

4.3 Evaluation metrics

In the experiments, we adopt two different metrics to evaluate the dialog and recommendation modules respectively, which are also common in previous work. Dialog evaluation includes automatic evaluation and human evaluation. For automatic evaluation, since the diversity score of vocabulary can measure the breadth and richness of words used in the text, this paper uses the diversity (Distinct) indicator to judge whether there are a large number of general and repetitive replies. Distinct is defined as follows:

$$\text{Distinct}(n) = \frac{\text{Count}(\text{unique } n\text{-gram})}{\text{Count}(\text{word})} \quad (6)$$

$\text{Count}(\text{unique } n\text{-gram})$ represents the number of unique n -grams present in the reply sentence, while $\text{Count}(\text{word})$ represents the total number of n -gram words in the reply sentence. In this paper, Distinct 2-gram, Distinct 3-gram, and Distinct 4-gram are used to evaluate the performance of the dialog module. The larger the Distinct- n , the higher the diversity of responses.

The evaluation of the recommendation module is based on the recall rate, which measures the ratio of relevant results retrieved from the top k items in the recommendation list to the total number of relevant items in the dataset. Recall rate ranges from 0 to 1, with higher values indicating better performance. We report the recall rates at different values of k , namely Recall@1, Recall@10, and Recall@50.

Meanwhile, for human evaluation, since we want the generated replies to be item-related suggestions, we invite five annotators with knowledge in linguistics and require them to score the generated sentences in two aspects, namely Fluency and Informativeness. The final performance is calculated using the average score of these annotators.

4.4 Compared methods

We compare our approach with models used by some mainstream dialog recommender systems:

TextCNN (Kim 2014): It proposed a CNN-based model for sentence-level classification tasks by extracting features in context. First, the natural language of the input sentence is encoded into a distributed representation through the embedding layer, and then, the different n -gram features of the sentence are extracted through a convolution layer, and finally, the output results are obtained through the fully connected layer.

Transformer (Wang *et al.* 2019): It adopted a transformer-based encoder-decoder framework consisting of self-attention and feed-forward neural networks to generate appropriate responses

Table 3. Results of recommendation system

Dataset	REDIAL			INSPIRED		
	Recall@1	Recall@10	Recall@50	Recall@1	Recall@10	Recall@50
TextCNN	1.3	6.8	19.2	–	–	–
ReDial	2.1	14.0	31.1	0.3	11.7	28.5
KBRD	3.1	15.1	33.6	5.8	14.6	20.7
KGSF	3.4	17.9	37.6	5.7	16.5	25.6
RevCore	4.1	18.7	37.6	05.3	0.157	24.8
C ² -CRS	4.9	23.0	40.2	6.5	18.2	28.3
Ours	3.7	19.9	39.0	6.4	18.0	27.8

for dialog modules. It can be trained in parallel, the speed is relatively fast, and the problem of long-distance dependence is well solved.

REDIAL (Li *et al.* 2018): This work is the earliest exploration of generative dialog recommendation to develop an agent that can chat with partners and ask their movie tastes in order to provide movie recommendations. It is mainly composed of HRED-based dialog generation system, auto-encoder-based recommender system, and sentiment analysis module.

KBRD (Chen *et al.* 2019): An end-to-end framework, KBRD, is proposed to bridge the gap between recommender systems and dialog systems through knowledge dissemination. External knowledge is also introduced to align entities that appear in the dialog content with nodes on the graph, making it possible for the system to recommend based on entities mentioned in the dialog.

KGSF (Zhou *et al.* 2020): It proposed a knowledge graph-based semantic fusion method that enhances the semantic representation of words and items and uses mutual information maximization to align the semantic spaces of two different components by using two external KGs.

RevCore (Lu *et al.* 2021): It performed comment enrichment and entity-based recommendations on item suggestions by extracting comments that match emotions and generating responses using a comment-focused encoder-decoder.

C²-CRS (Zhou *et al.* 2022): It proposed a new contrastive learning-based coarse-to-fine pre-training method, which effectively integrated multi-type data representations by adopting a coarse-to-fine pre-training strategy to enhance the representation of the conversation context.

KGTO (Pan, Yin, and Huang 2022): It presented a topic-oriented model based on keyword guidance, which used a hierarchical attention mechanism to extract keywords and capture more accurate topics. It also integrated co-occurrence and knowledge graphs to enrich contextual information of the item.

Among these baseline models, TextCNN is a recommendation method, transformer is the state-of-the-art text generation method. Redial is the earliest exploration of generative dialog recommendation. The rest Redial, KBRD, KGSF, C²-CRS, and KGTO are dialog recommender systems.

4.5 Results

In this subsection, we present our experimental results, including recommendation and dialog generation aspects. Table 3 shows the performance of different methods under three settings.

Table 4. Automatic evaluation of dialog system

Dataset	REDIAL			INSPIRED		
	Dist-2	Dist-3	Dist-4	Dist-2	Dist-3	Dist-4
Transformer	1.48	1.52	1.37	-	-	-
ReDial	2.25	2.40	2.29	0.406	1.226	2.205
KBRD	2.68	3.68	4.23	0.567	2.017	3.620
KGSF	2.89	4.35	5.29	0.608	2.519	4.929
RevCore	4.04	5.56	6.12	0.693	3.875	5.122
KGTO	3.67	4.72	5.25	0.667	3.426	4.791
C ² -CRS	1.63	2.91	4.17	0.522	2.117	3.980
Ours	3.99	5.79	6.83	0.734	4.267	5.340

4.5.1 Recommended module performance

To evaluate the effect of recommender systems, we evaluate Recall@K, which refers to whether the top k items selected by the recommender system contain ground truth recommendations provided by human recommenders. The results are shown in Table 3.

The results show that the CRS model is more effective than the simple recommendation method TextCNN. In contrast to the basic TextCNN, Redial uses entities or items in context to make recommendations, while KBRD and KGSF make more use of external knowledge. RevCore further enhances the generation of dialog responses by using emotionally coordinated comments on top of this. Our method is superior to all the above-generation methods. Unfortunately, our approach is worse than the C²-CRS model based on the pre-training approach. C²-CRS uses contrastive learning method to combine multi-type data such as text, item, and external comments related to entities to enhance data representation, which helps to better capture user preferences. In contrast, we only use words and items to align semantic representations, ignoring unstructured data such as reviews, and incomprehensively utilizing heterogeneous external data. However, this does not mean that our method is meaningless, as we can see below that our method performs better than C²-CRS in the dialog module, one possible reason is that the external comments introduced contain a lot of noise. In addition, we should note that our method outperforms all the generative methods. This shows that on the basis of the retrieval method, the introduction of knowledge graph and context based on time-series features can indeed accurately learn the feature relationship between users and items. Inspired by C²-CRS, we can study how to integrate more external data while improving the accuracy of recommendations and diversity of conversations.

4.5.2 Dialog module performance

In this subsection, we verify the effectiveness of the proposed model for the conversation task and describe the results on automatic and human evaluation metrics.

4.5.2.1 Automatic evaluation. Table 4 shows the evaluation results of the baseline models and the proposed methods in dialog generation. Dist-2, Dist-3, and Dist-4 represent distinct 2-gram, distinct 3-gram, and distinct 4-gram, respectively. We can see that ReDial performs better than transformer because ReDial applies RNN models to generate better historical dialog representations. Secondly, KBRD and KGSF enhance the context entities and items by external knowledge

Table 5. Human evaluation of dialog system

Model	Fluency	Info
Transformer	0.92	1.08
ReDial	1.37	0.97
KBRD	1.18	1.18
KGSF	1.50	1.41
RevCore	1.21	1.38
C ² -CRS	1.55	1.47
Ours	1.57	1.55

Table 6. Results of ablation analysis

Model	R@1	R@10	R@50	Dist-3	Dist-4
Ours	3.7	19.9	39.0	57.9	68.3
w/o Retrieval	3.5	19.5	38.5	55.0	64.7
w/o PE	3.6	19.3	38.2	53.6	63.4
w/o DTN	3.3	19.0	37.9	46.3	57.0
w/o DB	2.8	13.3	28.9	45.6	5.53

graphs, resulting in sentence diversity. Thirdly, RevCore enriches the dialog representation by introducing emotional factors, producing diverse responses. Additionally, KGTO and C²-CRS perform poorly, possibly due to focusing more on the accuracy of generating sentences, while neglecting to reduce the generation of safe responses. On the REDIAL dataset, our proposed method is close to the results of RevCore in the Dist-2 metric and significantly better than baselines in Dist-3 and Dist-4. On the INSPIRED dataset, our method also obviously outperforms the other models, indicating that introducing the temporal features of dialog content and retrieval model is extremely necessary. Our model can generate relatively more diverse content while ensuring fluency and informativeness.

4.5.2.2 Human evaluation. For human evaluation, we sample 100 multi-turn dialogs from REDIAL's test set together with the corresponding responses and require the annotators to score the whether the candidates have more meaningful information and fluency. The score range is 0–2 points, and we adopt the average score of these annotators. As shown in Table 5, compared with the best-performing C²-CRS model, our method achieved better performance with increases of 0.02 in fluency and 0.08 in informativeness, indicating that our model can effectively utilize context information and generate fluent and informative responses.

4.5.3 Ablation analysis

Additionally, this study also conducted ablation studies (as shown in Table 6) to observe the contribution of each component. Specifically, one module was removed for training, and then, the experimental results were compared. If the overall performance is lower after removing the module than before, it indicates that the module is effective in improving performance. Four ablation experiments were conducted in this paper: the method without the retrieval module

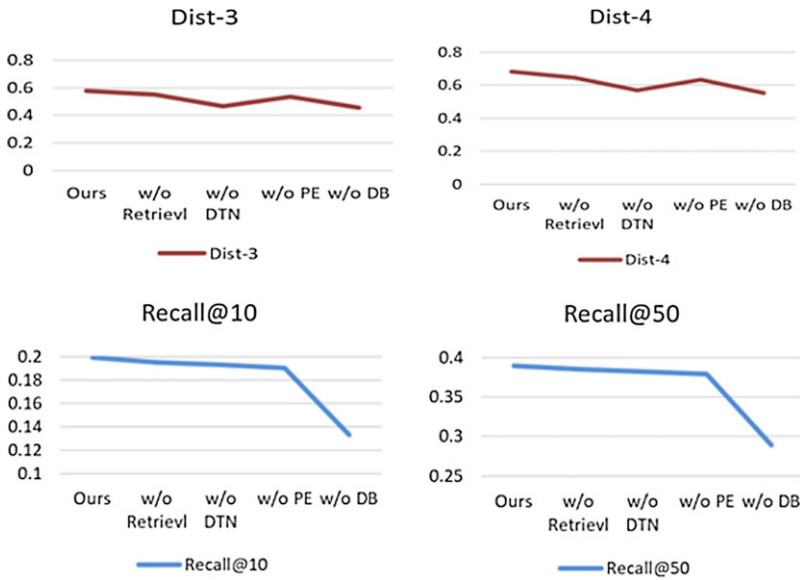


Figure 6. Line chart of ablation experiment results.

(w/o Retrieval), the method without positional encoding (w/o PE), the method without the proposed deep temporal network, and the method without knowledge graph (w/o DB).

From Table 6, we can observe that each indicator decreases to varying degrees when a particular part of the model is removed during training. Furthermore, from Fig. 6, it can be seen that the model performs best when the retrieval and generation modules are both present. The retrieval module provides accurate candidate replies for CRS and provides effective information for the generation model. When this part is removed, the system’s performance will decrease. Specifically, the knowledge graph provides various types of knowledge and background data for the dialog recommendation system. When the item-oriented KG is removed, the performance decrease is particularly significant. In addition, the DTN provides potential sequential dependencies for the dialog context. When the model does not add DTN, each indicator shows a significant decrease. The position encoding strengthens the importance of temporal features and reduces the occurrence of repeated words in the generated responses. When this part is removed, the performance also decreases accordingly. Therefore, we have reason to believe that the content of the dialog based on deep temporal features is particularly important in the dialog recommendation system, and the retrieval function is also indispensable, providing suitable recommendations together.

In conclusion, this study believes that the proposed method of fusing retrieval and generation based on time-series features can effectively enrich the performance of dialog recommendation systems. By combining the precision of retrieval methods with the fluency of generation methods, and utilizing the deep temporal features of dialog content, our method is highly effective.

5. Conclusion

This paper proposes an improved dialog recommender method based on time-series features of dialog context. By augmenting the semantic representation of word and item with two external knowledge graphs, the semantic space is aligned using mutual information maximization techniques. Additionally, we use a deep timing network model to provide different recommendations based on the order of dialog content and introduce the retrieved responses to provide the corresponding information for the generative model to improve the accuracy and diversity of the dialog

recommendation system. Through a series of experiments, we demonstrate that our method achieves good performance in both dialog and recommendation. In future work, we will focus on introducing more context-relevant external data and designing generic presentation models to incorporate the underlying semantics. We will also consider introducing new techniques to make conversations more persuasive and provide explanations for recommendations, while applying the model to more scenarios, such as multi-modal CRS.

Financial support. This work is supported by National Key Research and Development Program of China (2020AAA0109700), National Natural Science Foundation of China (62076167), the general project of the 14th Five-Year Scientific Research Plan of the National Language Commission (YB145-16) and China Post Doctoral Science Foundation (2022M722231).

References

- Bansal T., Belanger D. and McCallum A. (2016). *Ask the GRU: multi-task learning for deep text recommendations*. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 107–114.
- Bizer C., Lehmann J., Kobilarov G., Auer S., Becker C., Cyganiak R. and Hellmann S. (2009). DBpedia—a crystallization point for the web of data. *Journal of Web Semantics* 7(3), 154–165.
- Boussakssou M., Ezzikouri H. and Erritali M. (2022). Chatbot in Arabic language using seq to seq model. *Multimedia Tools and Applications* 81(2), 2859–2871.
- Cai D., Wang Y., Bi W., Tu Z., Liu X. and Shi S. (2019). *Retrieval-guided dialogue response generation via a matching-to-generation framework*. 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)*, pp. 1866–1875.
- Chen H., Liu X., Yin D. and Tang J. (2017). A survey on dialogue systems: recent advances and new frontiers. *ACM SIGKDD Explorations Newsletter* 19(2), 25–35.
- Chen Q., Lin J., Zhang Y., Ding M., Cen Y., Yang H. and Tang J. (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)*, pp. 1803–1813.
- Chen Y. (2015). *Convolutional Neural Network for Sentence Classification*. Master's Thesis, University of Waterloo.
- Chiang W.-L., Liu X., Si S., Li Y., Bengio S. and Hsieh C.-J. (2019). *Cluster-GCN: an efficient algorithm for training deep and large graph convolutional networks*. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 257–266.
- Christakopoulou K., Beutel A., Li R., Jain S. and Chi E.H. (2018). *Q&R: a two-stage approach toward interactive recommendation*. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 139–148.
- Christakopoulou K., Radlinski F. and Hofmann K. (2016). *Towards conversational recommender systems*. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 815–824.
- Divya M.S. and Goyal S.K. (2013). An advanced and quick search technique to handle voluminous data. *Compusoft* 2, 171–175.
- Feng Y., Lv F., Shen W., Wang M., Sun F., Zhu Y. and Yang K. (2019). Deep session interest network for click-through rate prediction. arXiv preprint [arXiv:1905.06482](https://arxiv.org/abs/1905.06482).
- Gao L., Wang J., Pi Z., Zhang H., Yang X., Huang P. and Sun J. (2020). A hybrid GCN and RNN structure based on attention mechanism for text classification. In *Journal of Physics: Conference Series*, vol. 1575. IOP Publishing, pp. 012130.
- Hayati S.A., Kang D., Zhu Q., Shi W. and Yu Z. (2020). INSPIRED: toward sociable recommendation dialog systems.
- He S., Liu C., Liu K. and Zhao J. (2017a). Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. In *ACL (1)*, pp. 199–208.
- He X., Liao L., Zhang H., Nie L., Hu X. and Chua T.-S. (2017b). *Neural collaborative filtering*. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 173–182.
- Jannach D., Manzoor A., Cai W. and Chen L. (2021). A survey on conversational recommender systems. *ACM Computing Surveys (CSUR)* 54(5), 1–36.
- Javed U., Shaukat K., Hameed I.A., Iqbal F., Alam T.M. and Luo S. (2021). A review of content-based and context-based recommendation systems. *International Journal of Emerging Technologies in Learning (IJET)* 16(3), 274–306.
- Kadlec R., Schmid M. and Kleindienst J. (2015). Improved deep learning baselines for ubuntu corpus dialogs. *Computer Science*. <https://doi.org/10.48550/arXiv.1510.03753>
- Kim Y. (2014). Convolutional neural networks for sentence classification. In *EMNLP*, pp. 1746–1751.
- Lei W., He X., Miao Y., Wu Q., Hong R., Kan M.-Y. and Chua T.-S. (2020). *Estimation-action-reflection: towards deep interaction between conversational and recommender systems*. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pp. 304–312.

- Li R., Ebrahimi Kahou S., Schulz H., Michalski V., Charlin L. and Pal C. (2018). Towards deep conversational recommendations. In *Advances in Neural Information Processing Systems* 31.
- Liu H., Lu J., Yang H., Zhao X., Xu S., Peng H., Zhang Z., Niu W., Zhu X., Bao Y. and Yan W. (2020). *Category-specific CNN for visual-aware CTR prediction at jd. com*. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2686–2696.
- Lowe R., Pow N., Serban I. and Pineau J. (2015). The Ubuntu dialogue corpus: a large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 285–294.
- Lu Y., Bao J., Song Y., Ma Z., Cui S., Wu Y. and He X. (2021). RevCore: review-augmented conversational recommendation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, pp. 1161–1173.
- Ma W., Takanobu R. and Huang M. (2021). CR-walker: tree-structured graph reasoning and dialog acts for conversational recommendation. In *Empirical Methods in Natural Language Processing*
- Manzoor A. and Jannach D. (2022). Towards retrieval-based conversational recommendation. *Information Systems* 109, 102083.
- Naous T., Hokayem C. and Hajj H. (2020). *Empathy-driven arabic conversational chatbot*. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pp. 58–68.
- Pan Y., Yin Y. and Huang F. (2022). *Keyword-guided topic-oriented conversational recommender system*. In *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–8.
- Polatidis N. and Georgiadis C.K. (2016). A multi-level collaborative filtering method that improves recommendations. *Expert Systems with Applications* 48, 100–110.
- Quadrana M., Karatzoglou A., Hidasi B. and Cremonesi P. (2017). *Personalizing session-based recommendations with hierarchical recurrent neural networks*. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pp. 130–137.
- Ricci F., Rokach L. and Shapira B. (2021). Recommender systems: techniques, applications, and challenges. In *Recommender Systems Handbook*, pp. 1–35.
- Shang L., Lu Z. and Li H. (2015). *Neural responding machine for short-text conversation*. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 1577–1586.
- Singla K., Chen Z., Atkins D. and Narayanan S. (2020). *Towards end-2-end learning for predicting behavior codes from spoken utterances in psychotherapy conversations*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3797–3803.
- Speer R., Chin J. and Havasi C. (2017). *ConceptNet 5.5: an open multilingual graph of general knowledge*. In *Proceedings of the AAAI conference on Artificial Intelligence*, vol. 31.
- Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł. and Polosukhin I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems* 30.
- Vinyals O. and Le Q. (2015). A neural conversational model. In *ICML Deep Learning Workshop*
- Wang H., Zhang F., Zhao M., Li W., Xie X. and Guo M. (2019). *Multi-task feature learning for knowledge graph enhanced recommendation*. In *The World Wide Web Conference*, pp. 2000–2010.
- Wang N., Wang S., Wang Y., Sheng Q.Z. and Orgun M. (2020). *Modelling local and global dependencies for next-item recommendations*. In *Web Information Systems Engineering–WISE 2020: 21st International Conference, Proceedings, Part II 21*, October 20–24, 2020, Amsterdam, The Netherlands. Springer, pp. 285–300.
- Woungang I., Dhurandher S.K., Pattanaik K.K., Verma A. and Verma P. (2023). *Advanced Network Technologies and Intelligent Computing: Second International Conference, ANTIC 2022, Proceedings, Part I*, December 22–24, 2022, Varanasi, India. Springer Nature.
- Yang L., Qiu M., Qu C., Guo J., Zhang Y., Croft W.B., Huang J. and Chen H. (2018). *Response ranking with deep matching networks and external knowledge in information-seeking conversation systems*. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 245–254.
- Yang Y., Tong Y., Ma S. and Deng Z.-H. (2016). *A position encoding convolutional neural network based on dependency tree for relation classification*. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 65–74.
- Yeh Y.T. and Chen Y.N. (2019). Qainfomax: learning robust question answering system by mutual information maximization.
- Yuan F., Karatzoglou A., Arapakis I., Jose J.M. and He X. (2019). *A simple convolutional generative network for next item recommendation*. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*, pp. 582–590.
- Zhang K., Qian H., Cui Q., Liu Q., Li L., Zhou J., Ma J. and Chen E. (2021). *Multi-interactive attention network for fine-grained feature learning in ctr prediction*. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pp. 984–992.

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

Zhou Y., Zhou K., Zhao W.X., Wang C., Jiang P. and Hu H. (2022). *C2-CRS: coarse-to-fine contrastive learning for conversational recommender system*. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pp. 1488–1496.