Complex & Intelligent Systems
https://doi.org/10.1007/s40747-024-01442-y

ORIGINAL ARTICLE

Multi-source information contrastive learning collaborative augmented conversational recommender systems

Huaiyu Liu[1] · Qiong Cao[1] · Xianying Huang[1] · Fengjin Liu[1] · Chengyang Zhang[1] · Jiahao An[1]

Received: 8 January 2024 / Accepted: 31 March 2024

Abstract

Conversational Recommender Systems (CRS) aim to provide high-quality items to users in fewer conversation rounds using natural language. Despite various attempts that have been made, there are still some problems: Previous CRS only learned item representations in a single knowledge graph and ignored item tags; information gaps exist in the same items from different knowledge graphs and information popularity both affect user preferences; system generated responses lack descriptiveness and diversity. To address these problems and fully utilize external knowledge, we propose a Multi-source Information Contrastive Learning Collaborative Augmented method (**MCCA**), which aims to mine the potential tag preferences of users in dialogues as well as enhance the accuracy of item representation and user preference modeling. Specifically, we utilize the obtained items and their tags to construct a new knowledge graph that incorporates movie tags. We design a Multi-source Item Fusion mechanism (**MIF**) to bridge the information gaps between items from different knowledge graphs and then utilize unsupervised contrastive learning to enhance the items' representation capability after MIF. Additionally, a Multi-Tag Fusion mechanism (**MTF**) is designed to combine user-perceived information (i.e., tag popularity) and keywords obtained from reviews to co-enhance user preference representations through items and tags, and to incorporate fused item and tag features into the conversation module. Extensive experiments on two datasets show that MCCA significantly outperforms state-of-the-art methods. The source code will be available at https://github.com/lhy-cqut/MCCA.

Keywords Conversational recommender systems · Knowledge graph · Tag information · Multi-source information · Contrastive learning

Introduction

Nowadays, interactive conversational intelligent assistants are developing rapidly, and people expect to obtain high-quality recommendations by engaging in natural language conversations with these intelligent assistants, which conversational recommender systems (CRS) [1, 2] are dedicated to doing. With the development of neural networks, CRS has been widely used in e-commerce, news, and other fields[3–8], demonstrating its value and drawing many researchers' interest.

Unlike traditional one-time recommender systems [9, 10], CRS interactively acquires user preferences through fewer conversation rounds to help users query information or accomplish specific tasks. Generally, CRS consists of two modules, the recommendation module and the conversation module [11, 12], the conversation module is responsible for the natural language conversation with users; the recommendation module is responsible for dynamically capturing user preferences and suggesting suitable items (movies). An example of CRS-user interaction is illustrated in Fig. 1. The process ends when the user accepts recommendations or exits

✉ Qiong Cao
jing5589@163.com

Huaiyu Liu
lhy_cqut@163.com

Xianying Huang
wldsj_cqut@163.com

Fengjin Liu
lfj_cqut@163.com

Chengyang Zhang
zcy_cqut@163.com

Jiahao An
ajh_cqut@163.com

[1] College of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054, China

Published online: 11 May 2024

Fig. 1 An example of movie recommendation conversation between user and system. Items and tags are marked in pink and yellow, respectively

the system. This interactive approach makes CRS more flexible and personalized, better meeting user needs.

In CRS, accurately capturing user preferences for recommendations is a challenging task, because it requires accurately capturing user preferences within relatively short multi-turn dialogues, however, the limited information contained in a few dialogues also increases the difficulty of the task. Since the number of items appearing in history conversations is very few, which causes the sparsity of the number of items, existing research mitigates this problem by introducing external knowledge (e.g., knowledge graphs, reviews, introductions, etc.) [1, 12–14], but some difficulties still remain.

In existing CRS the knowledge graph used is a subgraph of DBpedia, which ignores some information. Therefore, we introduced tags and constructed a new movie knowledge graph with tags. Tags are descriptions of the characteristics of items, generally condensed into phrases, they represent subjective evaluations generated by most people for a thing and can play a crucial role in providing recommendations to users.

However, due to data heterogeneity, directly utilizing external information from multiple sources to enhance CRS is difficult. This is because external information differs in content (a tag describes characteristics vs. a director is a person), and there is a natural information gap between them. Therefore, we must devise a method to bridge information gaps across different data sources. Besides, each tag varies in popularity among the general public (i.e., how many times the tag appears in all items), making it difficult for the system to effectively distinguish between popular and less popular tags. Popular tags are more likely to align with user preferences. Hence, we need to devise a method to address the issue of the effect of popularity on user preferences.

Furthermore, the responses generated by the system lack descriptiveness and diversity when making recommendations. As shown in Fig. 1, the user requests recommendations similar to "*Interstellar*", which includes tags such as "*science fiction drama films*", "*space adventure films*", etc., and the system's recommended items, "*The Martian*" and "*Passengers*", both match these tags. Thus, this recommendation better aligns with the user preferences, and the descriptiveness and diversity of the recommended items by the system can also be improved using tag information.

Therefore, we propose a new model, Multi-source Information Contrastive Learning Collaborative Augmented Conversational Recommender Systems(**MCCA**). Its core idea is to fully utilize multi-source content to improve the overall performance of the CRS. We first construct a new Movie Tag Knowledge Graph (**MTKG**), which contains association information between movies and their corresponding tags. Meanwhile, we combined DBpedia. To bridge the information gaps between the items in the two knowledge graphs, we devise a Multi-source Item Fusion mechanism (**MIF**). To fully mine the connection between different knowledge graphs and optimize the features of items that have been fused by MIF, we perform unsupervised contrastive learning on the fused item features separately with the item features from different knowledge graphs. Furthermore, we devise a Multi-Tag Fusion mechanism (**MTF**) to address the problem of different tag popularity, we obtain the keywords from the reviews of the corresponding items by the unsupervised keywords extraction algorithm YAKE [15] as auxiliary information to enhance the popularity of tag information. Thereby, we not only supplemented the item information but also enhanced the expression of the users' preferences. Using the fused items and tags in the conversation module can also help to make the generated responses richer. Extensive experiments based on baseline datasets indicate that MCCA outperforms the current state-of-the-art CRS models in both recommendation accuracy and response quality.

The contributions of this paper can be summarized as follows:

(1) We construct a tag-based knowledge graph to introduce tag information for user preference representation.
(2) We devise a Multi-source Item Fusion mechanism to bridge the information gaps between items from different sources and utilize unsupervised contrastive learning to optimize the fused item features.
(3) We devise a Multi-tag Fusion mechanism to address the effect of tag popularity differences on user preferences.
(4) Using tag information in the conversation module to improve the descriptiveness and diversity of dialogues.

Related work

In recent years with the rapid development of conversational recommender systems [12, 14, 16], it has become a hot topic

to have interactive conversations with users and dynamically obtain user intentions and preferences. CRS aims to talk with users and provide high-quality recommendations through natural language.

CRS

Current research in CRS can be categorized into two types: attribute-based and open-ended. Our research is based on open-ended CRS.

Attribute-based CRS focuses on asking users preference questions about item attributes to make recommendations [17–19]. This type of CRS relies on pre-defined rules (e.g., slot filling [20]) to interact with the user, and it focuses on completing recommendations in as few dialogue rounds as possible. Although such systems are easy to implement, they do not emphasize generating human-like natural language responses and therefore have a poor user experience.

Another type is open-ended CRS, which learns user preferences from the original dialogue and then generates responses resembling human dialogue by combining recommended content [21–23]. Due to the sparsity of data, most existing methods help CRS understand dialogues and capture user preferences by incorporating external knowledge. These external knowledge include entity-level knowledge graph [1, 17, 24, 25], word-level knowledge graph [13], reviews [14], and introductions [12]. To utilize this external knowledge, researchers propose to align different semantic spaces in the two knowledge graphs using mutual information maximization [13], and for external knowledge with different structures, researchers propose a contrastive learning framework ranging from coarse-grained to fine-grained [16] to further improve semantic fusion. Although these methods have improved the performance of CRS to some extent, it is still a challenge to utilize external knowledge more effectively.

Contrastive learning

Contrastive learning has been widely used computer vision [26] and information retrieval [27], demonstrating good results in both fields. It usually relies on data enhancement strategies such as image rotation, random cropping, etc. to generate a set of relevant positive samples for learning and negative samples from the dataset using random sampling. In the field of natural language processing, contrastive learning can be used to better align and ensure consistency in the semantic space [28], It can also be used to fuse a variety of information, such as knowledge graphs and text [16], text and images [29–31]. It enables different information to complement each other and fully exploit the potential of data.

Problem formulation

Open-ended CRS consists of recommendation module and conversation module, Formally, we denote a user by $u \in U$, for dialogue, we use $H = \{s_k\}_{k=1}^{n}$ to denote the list of utterances consisted of dialogue that has taken place between the user and the system, where s_k represents the utterance in the k-th turn of the dialogue, generated by either the user or the system. $\mathcal{I}$ denotes the whole set of items. For the knowledge graph DBpedia ($\mathcal{D}$), each knowledge fact is formatted as $\langle e_1, r_d, e_2 \rangle$, $e_1, e_2 \in \mathcal{E}$ denote entities, and $r_d \in \mathcal{R}_{\mathcal{D}}$ denotes the relationship between them. Given the dialogue history H, the goal of the recommendation module is to accurately capture the user's preference and generate a subset of candidate items $\mathcal{I}_{k+1} \in \mathcal{I}$ that satisfy this preference. Sometimes, $\mathcal{I}_{k+1}$ may be $\varnothing$. The purpose of the conversation module is to generate utterance s_{k+1} as a response, which may contain recommended items, if $\mathcal{I}_{k+1}$ is $\varnothing$, s_{k+1} is a chit-chat utterance. When the user accepts or exits the system, the whole process is over.

Methodology

In this section, we propose a MCCA method, which is overviewed in Fig. 2. There are four parts: multi-source information data, multi-source information fusion, conversation module, and recommendation module. In the multi-source information data part, we first utilize the newly constructed Movie Knowledge Graph, called MTKG (Sect. "MTKG composition"), to augment the user's representation of preferences on tags as well as the representation of items. Then, we retrieve all the tag information and review information about items that have been mentioned in the dialogue from the movie information database (Sect. "Multi-source information acquisition"). Next, we design a multi-source information fusion module (Sect. "Multi-source information fusion augmented by unsupervised contrastive learning"), which includes a fusion mechanism for multi-source item representations (Sects. "Multi-source item fusion mechanism","Item-level unsupervised contrastive learning") and a multi-tag fusion mechanism (Sect. "Multi-tag fusion mechanism")to enhance user representations.

Multi-source information data

MTKG composition

In past research, the recommendation module utilized only items that had been mentioned in a dialogue to represent user preferences. However, since only a few items appeared in the dialogue, this could lead to inaccuracies in capturing user preferences. In our approach, we introduce tags as external

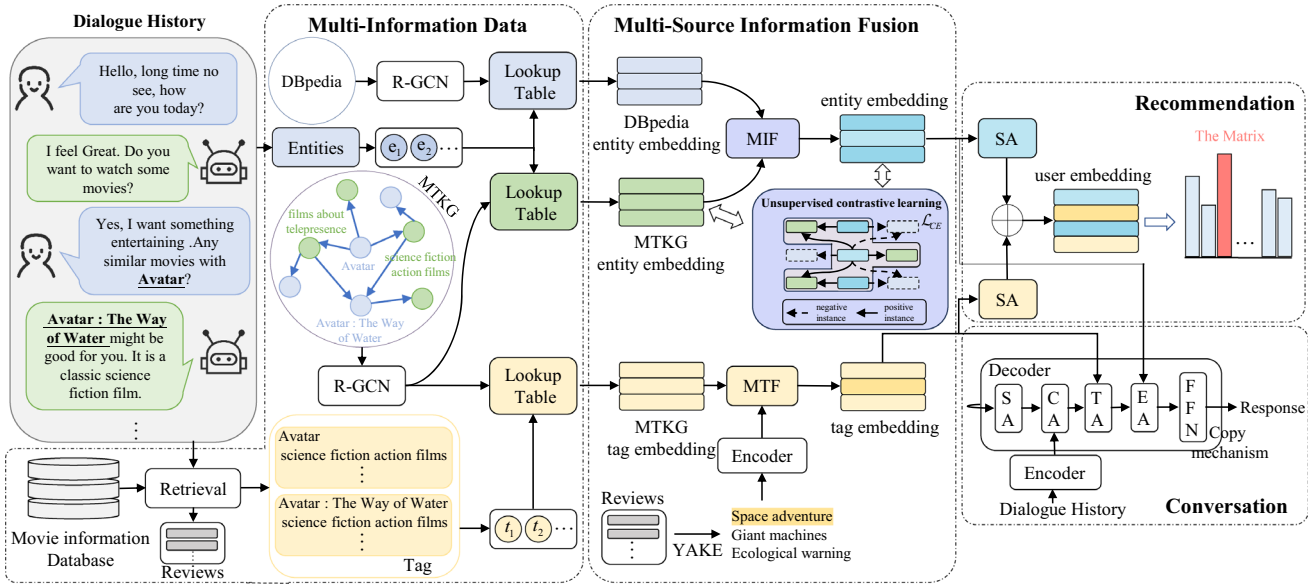

Fig. 2 Overview of the MCCA model. Bolded fonts in the context represent items. The yellow parts represent tag information or operations on tags, the green parts represent item information or operations on it in MTKG, and the blue parts represent item information or operations on it in DBpedia. MIF stands for Multi-source Item Fusion mechanism, and MTF stands for Multi-Tag Fusion mechanism

information which can highly summarize the main features of an item through phrases, thereby more accurately representing the characteristics of the items appearing in the dialogue, enabling a more precise capture of user preferences.

We introduced the existing knowledge graph DBpedia, in which the relationship between items and their tags was not stored. Therefore, we constructed a new movie knowledge graph with tag information.

Based on information about movie tags in Wikipedia, we collected tag information for all items that appeared in the dataset. A higher frequency of occurrence of a tag among all the tag information indicates a higher popularity of this tag among the people, and each movie corresponds to several tags, each of which is a phrase consisting of a few words, and these tags accurately describe the characteristic of the corresponding movie item. After obtaining the tag information for the items, we deleted some information that did not have special meanings (e.g., shot in 1988) and the tag information that appeared only once. All the items and corresponding tags collected after organization are saved in the database M_{db}, where we also store the collected tags' popularity information and the reviews of the items.

After processing the tag information, we made it into a new Movie-Tag Knowledge Graph called MTKG($\mathcal{G}$), In this knowledge graph, the relationship nodes between item entity nodes in this knowledge graph are tag information.

Multi-source information-aware item representation

We used the knowledge graph DBpedia to obtain the basic features representation of each item. We follow the existing approach of using R-GCN [14, 32] to encode each entity e in DBpedia($\mathcal{D}$). Formally, for each entity e in $\mathcal{D}$, its representation is computed by Eq. (1).

$$d_e^{\ell+1} = \sigma\left(\sum_{r\in\mathcal{R}_\mathcal{D}}\sum_{e\in\mathcal{E}_e^r}\frac{1}{Z_{e,r}}W_{d,r}^{\ell}h_e^{\ell} + W_d^{\ell}h_e^{\ell}\right) \tag{1}$$

where $h_e^{\ell} \in \mathbb{R}^d$ is the entity representation of e at ℓ-th layer, d denotes the feature embedding dimension, $\sigma(\cdot)$ denotes the $ReLU$ activation function, $\mathcal{E}_e^r$ denotes the set of neighboring nodes of e under the r relation, $W_{d,r}^{\ell}$, W_d^{ℓ} are the learnable matrixs, and $Z_{e,r}$ is the normalization factor. According to Eq. (1), each node can aggregate information from different entity nodes through messaging, and we use its output $D_\mathcal{D} = \{d_{e,1}, d_{e,2}, \cdots, d_{e,m}\}$ as the foundational representation for items, where m denotes the number of items.

To obtain embeddings of each item and tag in the knowledge graph MTKG($\mathcal{G}$), we use another R-GCN to encode them.

$$g_i^{\ell+1} = \sigma\left(\sum_{r\in\mathcal{R}_\mathcal{G}}\sum_{j\in\mathcal{G}_i^r}\frac{1}{Z_{i,r}}W_{g,r}^{\ell}h_j^{\ell} + W_g^{\ell}h_i^{\ell}\right) \tag{2}$$

where $g_i^{\ell} \in \mathbb{R}^d$ represents the node's representation at layer ℓ-th, where d is the embedding dimension, $\sigma(\cdot)$ is the $ReLU$ activation function, $\mathcal{G}_i^r$ is the set of neighboring nodes of the node under the relation r, $W_{g,r}^{\ell}$, W_g^{ℓ} are learnable matrixes, and $Z_{i,r}$ is the normalization factor. Similarly, according to Eq. (2) we can obtain the embedding representation in MTKG, and we take its output $D_{\mathcal{G}} = \{g_{i,1}, g_{i,2}, \cdots, g_{i,n}\}$ as the base representation and n is the number of nodes.

Multi-source information acquisition

First, extract the items E_h from the the dialogue history H. Extraction using a character matching model, the process of extracting the items is shown in Eq. (3).

$$E_h = Extract(H) \tag{3}$$

where $Extract(\cdot)$ denotes the extract operation.Then, all the tags, the tags' popularity, and reviews corresponding to the items are retrieved from the movie information database M_{db} based on the items appearing in the dialogue history H. The retrieval is calculated as shown in Eq. (4).

$$T_{ment}, T_{pop}, R = Retrieve(E_h, M_{db}) \tag{4}$$

where $Retrieve(\cdot)$ denotes the retrieval operation, T_{ment} denotes a set of retrieved tags, T_{pop} denotes a set of retrieved tags' popularity, and R denotes a set of retrieved reviews.

After obtaining the information above, querying the embedding dictionary $D_{\mathcal{D}}$ of the knowledge graph DBpedia obtained in Sect. ("Multi-source information-aware item representation") obtains the items' embedding set E_d; querying the embedding dictionary $D_{\mathcal{G}}$ of the MTKG obtains the items' embedding set E_g and the tags' embedding set E_t, and then the items' embedding vectors are respectively spliced to form the items' feature embedding matrixes $E_{\mathcal{D}} \in \mathbb{R}^{a \times d}$ and $E_{\mathcal{G}} \in \mathbb{R}^{a \times d}$, similarly, the tags' embedding vectors are spliced to form the tags' feature embedding matrix $E_{\mathcal{T}} \in \mathbb{R}^{b \times d}$, where a and b denote the number of items and tags, respectively.

Multi-source information fusion augmented by unsupervised contrastive learning

Multi-source item fusion mechanism

In Sect. ("Multi-source information acquisition"), we obtained the item embedding matrices $E_{\mathcal{D}}$ and $E_{\mathcal{G}}$. For a certain item obtained from different knowledge graphs, they represent the same item in terms of information (e.g., "*Inception*" in DBpedia vs. "*Inception*" in MTKG), simply concatenating the embeddings from two different sources does not fully utilize the potential of the data.

In order to bridge the information gaps exhibited between items from different knowledge graphs, we designed a Multi-source Item Fusion mechanism (**MIF**) to solve this problem. MIF is used to align informationally highly correlated two item embeddings and fuse them. The core idea is to utilize the similarity between two coupled variables to solve the problem of information gaps, which is calculated as follows in Eq. (5).

$$sim\left(E_{\mathcal{D}_i}, E_{\mathcal{G}_j}\right) = \sigma\left(W_2 \cdot \sigma\left(W_1 \cdot \left[E_{\mathcal{D}_i}; E_{\mathcal{G}_j}\right] + b_1\right) + b_2\right) \tag{5}$$

where $sim(\cdot)$ represents the similarity score between two embeddings of a certain item with consistent informativeness from different sources, $E_{\mathcal{D}_i}$, $E_{\mathcal{G}_j}$ represent two embeddings of an item from two knowledge graphs with the same informativeness, respectively. W_1 and W_2 are matrices of learnable parameters, $[;]$ denotes the concatenation operation, and b_1, b_2 are the two bias terms. $\sigma(\cdot)$ denotes the $ReLU$ activation function. After obtaining the similarity, we fused the item embeddings to obtain the final item embedding E_{fusion}, with the following formula:

$$\begin{aligned} s_{i,j} &= \frac{exp\left(sim\left(E_{\mathcal{D}_i}, E_{\mathcal{G}_j}\right)\right)}{\sum_{k=1}^{n} exp\left(sim\left(E_{\mathcal{D}_i}, E_{\mathcal{G}_j}\right)\right)} \\ E_{fusion} &= \sum_{i=1}^{n}\sum_{j=1}^{n} s_{i,j} \cdot E_{\mathcal{D}_i} + \sum_{i=1}^{n}\sum_{j=1}^{n} s_{i,j} \cdot E_{\mathcal{G}_j} \end{aligned} \tag{6}$$

We apply $softmax$ to obtain the attention weights $s_{i,j}$. Finally, each item feature embedding $E_{\mathcal{D}_i}$ and $E_{\mathcal{G}_j}$ are weighted and fused with the attention weights to obtain the fused item feature embedding E_{fusion}.

Item-level unsupervised contrastive learning

A better embedding of fused item features was obtained through the MIF module, but the connection between the fused item features and the item features in each information source was not fully mined. We repeatedly perform mutual information maximization between the input features of each source item and the fused item features to optimize the fusion network from each source item to the fused item features. In the above sections, we have obtained item features representations $E_{\mathcal{D}}$, $E_{\mathcal{G}}$ from different sources and fused item features representation E_{fusion}. But there is a lack of mining the connection between the fused item features E_{fusion} and each information source E_x, $x \in \{\mathcal{D}, \mathcal{G}\}$. We measure the connection between them by using a score function $Score(\cdot)$ with normalized predictions and true vectors according to the operation of Vinyals et al. [33].

$$\begin{aligned} Score(E_x, E_{fusion}) &= exp(\overline{E}_x(\overline{G}_{\varphi}(E_{fusion}))^T) \\ \overline{G}_{\varphi}(E_{fusion}) &= \frac{G_{\varphi}(E_{fusion})}{\|G_{\varphi}(E_{fusion})\|_2}, \overline{E}_x = \frac{E_x}{\|E_x\|_2} \end{aligned} \tag{7}$$

where $\overline{G}_{\varphi}$ is a neural network with parameter φ that generates a prediction for E_x from E_{fusion}, and $\|\cdot\|_2$ is the Euclidean norm. We consider the other source features of this information in the same batch as negative samples, thus yielding the loss between the different source item features and the fused item features as shown in Eq. (8).

$$\mathcal{L}\left(E_{fusion}, E_x\right) = -\mathbb{E}_s\left[\log \frac{\text{Score}\left(E_{fusion}, e_x^i\right)}{\sum_{e_x^j \in E_x} \text{Score}\left(E_{fusion}, e_x^j\right)}\right] \tag{8}$$

The final loss for item-level unsupervised contrastive learning is obtained in Eq. (9).

$$\mathcal{L}_{CE} = \mathcal{L}(E_{fusion}, E_{\mathcal{D}}) + \mathcal{L}(E_{fusion}, E_{\mathcal{G}}) \tag{9}$$

Through item-level unsupervised contrastive learning, we can optimize the fusion parameters and obtain fused item features E_{fusion} that better fuse multi-source item features.

Multi-tag fusion mechanism

We enhance the representation of user preferences from an item perspective, and at the same time, we further enhance the user preference representations by using user-perceived information to better capture user preferences. In real life, things that appear more frequently tend to be more popular. Similarly, items corresponding to tags that appear more frequently should have more audience, that is, the more popular this tag information is the more popular the item with this tag is. In our collected data, each item is described by multiple tags, but the popularity of each tag is different, which results in different tags having different degrees of importance. In order to emphasize the popularity of different tags, we also utilized keywords extracted from the reviews to assist in enhancing the popularity of tag information. We designed a Multi-Tag Fusion mechanism (**MTF**) to handle the problem of the different popularity of tags.

The tag embedding representation $E_{\mathcal{T}}$ for all items that have appeared in the dialogue history is obtained in Sect. ("Multi-source information acquisition"). We use the keywords extracted by YAKE [15] from the reviews corresponding to all the items, and we want to obtain keywords that are more relevant to our dialogue, so we calculate the similarity scores between the keywords and the dialogue, we select the keywords with the highest similarity score as the auxiliary enhancement information to the tag information.

$$\begin{aligned} K &= Top{-}1\left(YAKE(R), H\right) \\ T_p &= Top{-}4\left(E_{\mathcal{T}}, T_{pop}\right) \end{aligned} \tag{10}$$

where Top–1 $(\cdot)$ denotes the select operation with the highest similarity score, $YAKE(\cdot)$ denotes the YAKE approach, K is the selected keywords, $Top-4\,(\cdot)$ denotes the operation of selecting the four tags with the highest popularity among all the tags, and T_p denotes the final tag feature embedding.

For the obtained keyword K is encoded using a Transformer encoder [34] to obtain $\mathcal{K}$, which is then fused with the tag feature to get the final tag feature $E_{Tfusion}$. The calculation process is shown in Eq. (11).

$$\begin{aligned} Pop_i &= \frac{exp\left(\frac{T_{p_j} \cdot \mathcal{K}}{\|T_{p_j}\| \|\mathcal{K}\|}\right)}{\sum_{j=1}^{n} exp\left(T_{p_j}\right)} \\ E_{Tfusion} &= \sum_{i=1}^{n}\left(Pop_i \cdot T_{p_i}\right) \end{aligned} \tag{11}$$

where Pop_i denotes the final tag popularity and $E_{Tfusion}$ denotes the final tag feature embedding after fusion with tag feature embedding T_{p_i}.

Multi-source information collaborative augmented recommendation module

Past research on the recommendation module has mainly used items that have appeared in history dialogue to represent user preferences, but user preferences captured through such an approach are often inaccurate because the number of items that can appear in history dialogue is very few, which results in sparsity in the number of items and hence inaccuracy in capturing user preference.

The fused item feature embedding E_{fusion} and tag feature embedding $E_{Tfusion}$ obtained above are applied self-attention mechanism to aggregate the item feature embedding and tag feature embedding, respectively, and then these two are finally fused using the gating mechanism to obtain the user preference embedding. The specific calculation process is shown in Eq. (12).

$$\begin{aligned} u^{(e)} &= SA\left(E_{fusion}\right) \\ u^{(t)} &= SA\left(E_{Tfusion}\right) \\ u^{(et)} &= \beta \cdot u^{(e)} + (1-\beta) \cdot u^{(t)} \end{aligned} \tag{12}$$

where $SA\,(\cdot)$ denotes the self-attention aggregation operation, $u^{(e)}$, $u^{(t)}$ denote the preference embedding representation of user preference at item level and tag level, respectively, and β denotes the gating probability. Ultimately, based on the obtained user preference embedding $u^{(et)}$, we can compute the probability $P_{rec}\,(j)$ of recommending item j to user u from the item set $\mathcal{I}$.

$$P_{rec}\,(j) = softmax\left(MLP\left(u^{(et)}\right)\right) \tag{13}$$

where $MLP(\cdot)$ denotes the multi-layer perceptron. Based on existing work, we learn the model parameters with cross-entropy loss and unsupervised contrastive learning loss as

optimization objectives.

$$\begin{aligned}\mathcal{L}_C &= -\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij}\cdot log\left(P_{rec}^{i}(j)\right)\\ \mathcal{L} &= \mathcal{L}_C + \omega\mathcal{L}_{CE}\end{aligned} \tag{14}$$

where $\mathcal{L}_C$ is the cross-entropy loss, $\mathcal{L}_{CE}$ is the unsupervised contrastive learning loss, N is the total number of conversations, i is the current conversation index, M is the number of items, and j is the index of an item, $\mathcal{L}$ is the sum of cross-entropy loss and unsupervised contrastive learning loss. ω is the weight of unsupervised contrastive learning loss.

Multi-source information collaborative augmented conversation module

The multi-source information collaborative augmented conversation module can not only generates chit-chat to explore user preferences but can also generates statements containing recommended items to recommend to the user. To improve the descriptiveness and diversity of the system-generated dialogue, tag information is introduced for further description of the recommended items.

We adopt the widely used language generation model Transformer [34] as the backbone model for response generation, which has shown excellent results in many natural language processing tasks [35–38]. We first encode the dialogue history H using the Transformer encoder to obtain a hidden representation $\mathcal{H}$ of the dialogue history. Then, we employ a transformer decoder with multi-head attention mechanism to gradually fuse context information, tag feature embedding and item feature embedding. The process is shown in Eq. (15).

$$\begin{aligned}A_0^n &= MHA\left(Y^{n-1}, Y^{n-1}, Y^{n-1}\right)\\ A_1^n &= MHA\left(A_0^n, \mathcal{H}, \mathcal{H}\right)\\ A_2^n &= MHA\left(A_1^n, E_{Tfusion}, E_{Tfusion}\right)\\ A_3^n &= MHA\left(A_2^n, E_{fusion}, E_{fusion}\right)\\ Y^n &= FFN\left(A_3^n\right)\end{aligned} \tag{15}$$

where Y^{n-1} represents the output of the decoder at time step $n-1$, E_{fusion}, $E_{Tfusion}$ are the item feature embedding and tag feature embedding from the multi-source information fusion module, respectively, and A_0^n, A_1^n, A_2^n, A_3^n represent the embedding that is output after the self-attention layer and the multi-information cross-attention layer, respectively. $MHA(Q,K,V)$ denotes the multi-head attention function [34], which was calculated as follows in Eq. (16).

$$\begin{aligned}MHA(Q,K,V) &= [h_1;\ldots;h_h]W^o\\ h_i &= Attention\left(QW_i^q, KW_i^k, VW_i^v\right)\end{aligned} \tag{16}$$

where Q query matrix, key matrix K and value matrix V are the inputs, h is the number of heads and W_i is the parameter matrix. $FFN(\cdot)$ is a fully connected feedforward neural network, consisting of two linear layers with $ReLU$ activation function.

$$FFN(x) = ReLU\left(xW_3 + b_3\right)W_4 + b_4 \tag{17}$$

where W_3, W_4 are learnable parameters and b_3, b_4 are two biases.

In order to generate dialog utterances, the decoder output Y_n also needs to predict the distribution of words through the $softmax$ operation. The CRS wants to incorporate the relevant content of the recommendation module when generating the response, as well as to provide a general description of the recommended items. The copy mechanism [39] achieves the above purposes and increases the information content and richness of the response. For a given generated sequence $\{y_{i-1}\} = y_1, y_2, \ldots, y_{i-1}$, the generate next marker y_i probability is shown in Eq. (18)

$$\begin{aligned}Pr\left(y_i|\{y_{i-1}\}\right) = Pr_1\left(y_i|Y_i\right) + Pr_2\left(y_i|Y_i, E_{Tfusion}\right)\\ +Pr_3\left(y_i|Y_i, E_{fusion}\right)\end{aligned} \tag{18}$$

Where Y_i is the output of the decoder, $Pr_1(\cdot)$ is the probability function for generating ordinary words from the vocabulary, and $Pr_2(\cdot)$, $Pr_3(\cdot)$ are the probability functions for words from tag and entity, respectively. We set the cross-entropy loss to optimize the generation of responses for the conversation module.

$$\mathcal{L}_{gen} = -\frac{1}{N}\sum_{t=1}^{T} log\,Pr\left(y_i|\{y_{i-1}\}\right) \tag{19}$$

Where T is the number of turns in dialogue and s_t is the t-th sentence in the dialogue.

Experiments

In this section, we will introduce the datasets, the baseline model, the evaluation metrics, and the implementation details.

Datasets

In the CRS domain, two English conversation datasets, REDIAL [40] and INSPIRED [41], are widely used.

The REDIAL dataset is constructed by Amazon Mechanical Turk (AMT) with conversations centered around movies in which one party seeks recommendations (the user) and

the other provides them (the system), and contains a total of 10,006 dialogues with a total of 182,150 sentences, involving 956 users and 51,699 movies.

The INSPIRED dataset is also an English dataset about movies, but it is a smaller dataset, containing 1,001 dialogues with a total of 35,811 sentences about 1,783 movies. Table 1 summarizes the statistics of the two datasets after processing.

Baseline model

We compared our approach to several competitive baseline models, including:

- **Redial** [40]: It is a baseline model published on the REDIAL dataset and consists of two components: an autoencoder-based recommendation module and a response generation module using HRED.
- **KBRD** [1]: The model utilizes DBpedia to augment the representation of items in context, then uses the Transformer architecture and incorporates information from the Knowledge Graph as a word bias for the response statements.
- **KGSF** [13]: The model utilized two knowledge graphs, one based on the word level and the other on the item level. Align the semantic space of different knowledge graphs by mutual information maximization.
- **RevCore** [14]: The model enhances the representation of items in context by introducing review information and making the generated responses more diverse.
- **C^2-CRS** [16]: The model uses Contrastive Learning to learn about data signals at different granularities to better fuse user preferences.
- **LOT-CRS** [42]: The model has solved the long-tail problem in the CRS dataset and improved the recommendation performance.

In the above baseline Redial, KBRD, KGSF, RevCore, C^2-CRS, LOT-CRS are all the models for conversational recommender systems.

Evaluation metrics

When evaluating the recommendation module and the conversation module, we need to use different evaluation metrics.

For the recommendation module, we aim to verify that our model accurately captures user preferences and provides high-quality recommendations. Therefore, we evaluate the recommendation task using Recall@k and MRR@k (k=1,10,50), which are used to evaluate whether the top-k recommended items generated by the model contain the real item labels provided by the dataset.

For the conversation module, we employed two evaluation methods: automatic evaluation and human evaluation. In the automatic evaluation, we used Perplexity [43] and Distinct-n (n=2,3,4) [44] to evaluate the dialogue quality, Perplexity measures the fluency of response generation, with lower Perplexity values implying more fluent sentences. Distinct-n is used to measure the diversity of generated responses, where n represents n-grams, indicating the diversity of combinations of consecutive n words considered in a sentence. A higher Distinct value indicates that the generated responses are more diverse. In the human evaluation, the evaluator scores both sentence fluency and informativeness with a range of scores [0,2], The average of the evaluator's scores is used as the human evaluation result to assess the quality of the dialog more comprehensively.

Implementation details

We implemented our MCCA model based on PyTorch and trained it on NVIDIA GeForce RTX 3090. We set the maximum length of the dialog context to 256, and the hidden dimensions of the recommendation and dialog modules to 128 and 300, respectively. To balance efficacy and efficiency, we employed two one-layer R-GCNs to encode two different knowledge graphs, with the normalization factor of R-GCN set to the default value of 1.0. The coefficient ω for unsupervised contrastive learning loss is 0.05. We will first conduct pre-training for 3 epochs, and then proceed with training the recommendation module and the conversation module separately. During the training process, we utilize the Adam optimizer [45], with a batch size of 128 and an initial learning rate set to 0.001. Additionally, we employ gradient clipping strategy to limit the gradients within the range of [0, 0.1]. To ensure fair comparison and achieve optimal performance, the baselines' hyperparameter settings follow their respective implementations.

Results and analysis

In this section, we verify the validity of our model through experiments and analyze the case.

Evaluation of recommendation module

Results analysis

Table 2 shows the results of different methods on the recommendation task, KBRD introduces knowledge graph to enhance item representation, KGSF makes significant progress on KBRD by using mutual information maximization for its item-level and word-level semantic space, RevCore enhances item representation by introducing

Table 1 The REDIAL dataset vs. the INSPIRED dataset

Dataset	Language	Domain	#Dialogue			#Sentences			#Items		
			Train	Train	Test	Train	Train	Test	Train	Train	Test
REDIAL	English	Movie	9006	901	1342	145,186	18,482	23,952	55,710	6059	8226
INSPIRED	English	Movie	801	99	99	16,982	2053	2089	3502	479	415

Table 2 Results of the recommendation task. We simplify Recall@k as R@k and MRR@k as M@k, respectively

Model	REDIAL					INSPIRED				
	R@1	R@10	R@50	M@10	M@50	R@1	R@10	R@50	M@10	M@50
ReDial	0.024	0.140	0.320	0.071	0.074	0.031	0.117	0.285	0.064	0.073
KBRD	0.031	0.150	0.336	0.075	0.078	0.058	0.146	0.207	0.077	0.081
KGSF	0.039	0.183	0.378	0.080	0.088	0.058	0.165	0.256	0.087	0.089
RevCore	0.046	0.220	0.396	0.082	0.090	0.068	0.198	0.379	0.087	0.094
C^2-CRS	0.050	0.233	0.407	0.083	0.092	0.090	0.242	0.399	0.101	0.096
LOT-CRS	0.049	0.216	0.416	0.077	0.082	0.094	0.250	0.410	0.109	0.117
MCCA	**0.056**	**0.259**	**0.456**	**0.087**	**0.095**	**0.104**	**0.262**	**0.443**	**0.114**	**0.126**

Bold indicates the best result.

reviews, C^2-CRS improves the effect by better integrating user preferences through coarse-to-fine contrastive learning, and LOT-CRS solves the long-tailed problem of the dataset to make the model more effective and performs better than other baseline models. The effects brought about by these approaches are progressively enhanced, in terms of performance, the order is ReDial<KBRD<KGSF<RevCore<C^2-CRS<LOT-CRS.

According to the results in Table 2 what we can see is that our model outperforms all other baseline models. In terms of introducing external knowledge, we construct a new movie knowledge graph, MTKG, and introduce both item and tag information, at the same time, our knowledge graph is only for the movie domain, reducing other domain noise interference. In terms of enhancing user preference representation, we fully mine the potential tag preferences in the dialogue and design an item-level information fusion mechanism and a tag-level information fusion mechanism to enhance user preference representation, while fully mining the inter-item connections using unsupervised contrastive learning to improve the CRS performance. Compared with LOT-CRS, our model has shown an improvement of 14.2% in R@1 and 19.9% in R@10 on the REDIAL dataset. At the same time, there was a significant improvement in the MRR metric as well. On the INSPIRED dataset, there is an improvement of 10.6% in R@1 and 4.8% in R@10. However, the improvement on the INSPIRED dataset is not as significant as that on the REDIAL dataset, possibly due to the smaller size of the INSPIRED dataset. Overall, our model shows a noticeable improvement on both datasets, this showed that the new knowledge graph we constructed and the multi-source information fusion mechanism can effectively help the model accurately capture user preferences.

Ablation study

In the recommendation section, we obtain embedding representations of items and tags from the high-quality movie knowledge graph MTKG we constructed. We fused the MTKG item embedding with those extracted from the DBpedia and performed unsupervised contrastive learning between the fused item embedding and embeddings from different knowledge graphs. Additionally, we perform multi-tag fusion for tag embedding to integrate user preferences from multiple sources, enhancing the model's recommendation performance.

In order to validate the effectiveness of our approach, we conducted a series of ablation experiments, in which we focused on the newly added components, (1) Removing the unsupervised contrastive learning module (w/o UCL); (2) Removing multi-source item fusion mechanism (w/o MIF); (3) Removing multi-tag fusion mechanism (w/o MTF); 4) Removing tag embedding (w/o Tag); 5) Removing MTKG item embedding (w/o M entity), i.e., use only item representations extracted from DBpedia; (6) Removing DBpedia item embedding (w/o D entity), i.e., use only the item representation extracted from MTKG.

Based on the results of the ablation experiments in Fig. 3, we observed that each component plays an important role in improving the accuracy of the recommendations. Particularly, among all the metrics, the performance of "w/o UCL", and "w/o MIF" decreased dramatically, such results indicate that the fusion of knowledge graphs from different

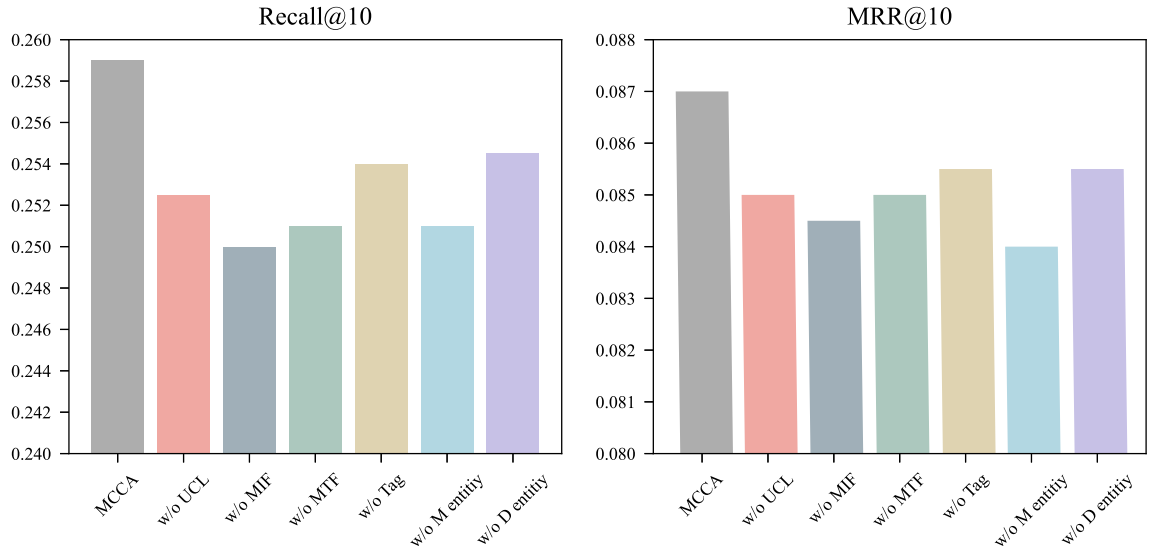

Fig. 3 Results of ablation experiments for Recall@10 and MRR@10 on the REDIAL dataset for recommended tasks

Table 3 The results of the automatic evaluation of the conversation task. We simplify Distinct-n as Dist-n and Perplexity as PPL

Model	REDIAL				INSPIRED			
	Dist-2	Dist-3	Dist-4	PPL	Dist-2	Dist-3	Dist-4	PPL
ReDial	0.226	0.238	0.230	28.1	0.406	1.226	2.205	30.2
KBRD	0.261	0.366	0.421	17.7	0.567	2.017	3.621	16.3
KGSF	0.289	0.431	0.520	11.8	0.608	2.519	4.929	10.5
RevCore	0.422	0.556	0.613	10.2	2.619	4.010	5.145	13.1
C^2-CRS	0.413	0.619	0.767	9.7	2.556	4.445	5.194	10.7
LOT-CRS	0.488	0.608	0.792	10.1	3.039	4.657	5.635	10.3
MCCA	**0.537**	**0.662**	**0.853**	**9.7**	**3.311**	**4.986**	**6.154**	**10.1**

Bold indicates the best result

sources is essential because they contain rich information about the items and there are information gaps, and bridging them allows for a more effective representation of user preferences. In addition, the "w/o MTF" metrics decreased, suggesting that tag information popularity benefits the system in enhancing item representations and making more accurate recommendations. The sharp decrease in "w/o M entity" compared to "w/o D entity" indicates that there is less noise in the MTKG, and reducing the noise also improves the model.

Evaluation of conversation module

Results analysis

We evaluated the conversation task using both automatic and human evaluation. Tables 3, 4 show the results of the different evaluation methods on the conversation task, compared according to the corresponding metrics, and we highlight the best results in bold. Through automatic evaluation, the KBRD and KGSF models perform better, indicating that external knowledge and semantic similarity information contribute to generating better responses. The RevCore model further improves the performance of the conversation module by introducing reviews. C^2-CRS better incorporates user preferences through multi-granularity Contrastive learning. LOT-CRS solves the long-tailed problem in the CRS dataset, further improving the performance of CRS.

Our model introduced tag information through the MTKG and employed unsupervised contrast learning to further enhance the item representations and improve the descriptiveness and diversity of system-generated responses. In terms of automatic evaluation, on the REDIAL dataset, our model achieved a significant improvement in automated assessment compared to the baseline model LOT-CRS, with Dist-2 and Dist-3 metrics improving by 10.0% and 8.8%, respectively. On the INSPIRED dataset, the Dist-2 and Dist-3 metrics improved by 8.9% and 7.0%, respectively. The results on both datasets show that our introduction of tag information is effective. In terms of human evaluation, the results in Table 4 show that our model also outperforms response fluency and diversity.

Table 4 Results of human evaluation for the conversation task.

Model	Fluency	Informativeness
ReDial	1.21	0.89
KBRD	1.24	1.16
KGSF	1.47	1.31
RevCore	1.51	1.37
C^2-CRS	1.54	1.42
LOT-CRS	1.51	1.49
MCCA	**1.56**	**1.53**

Bold indicates the best results

Table 5 Ablation experiments on the REDIAL dataset regarding the conversation task

Model	Dist-2	Dist-3	Dist-4
MCCA	**0.537**	**0.662**	**0.853**
w/o F tag	0.504	0.614	0.808
w/o F entity	0.522	0.636	0.826

Ablation study

In order to verify the effectiveness of our model in the conversation task, we conducted ablation experiments, as shown in Table 5. "w/o F tag" means removing the fused tag embedding. "w/o F entity" implies the use of non-fused item embeddings, where only the item embeddings are spliced. The results of the ablation experiment "w/o F tag" show that movie tags can enhance the diversity of system responses. The "w/o F entity" results demonstrate that fusing items from various sources can improve the performance of the system's production response.

Hyperparameter study

Effect of number of tags

In our approach, each movie corresponds to multiple tags, so the number of tags obtained is different. In order to explore the effect of the number of tags for each movie on the performance of the model, we conducted a series of experiments on the number of movie tags. As shown in Fig. 4a, we set the number of tags for each movie to 1, 2, 3, 4, 5, and 6. Our model shows the best performance on the recommendation task when the number of movie tags is set to 4. Therefore, we can observe that including rich tag information can improve the accuracy of the model in capturing user preferences, but if too many tags are included (e.g., when the number of tags is set to 5 or 6) too much noise is introduced, affecting the model's ability to capture user preferences, and thus reduces the model's recommendation performance.

Effect of weights

For the selection of weight ω for the loss of unsupervised contrastive learning in the recommendation task, we analyzed Recall@10 for the recommendation task on the REDIAL dataset with other parameters fixed. The results are presented in Fig. 4b. From the figure, it can be observed that the model shows the best performance when ω is set to 0.1. This means that the full use of information from different sources can indeed lead to a more accurate representation of user preferences, and setting an appropriate weight in the loss function can make better use of multi-source information and enable the model to better capture user preferences.

Effect of the number of R-GCN layers

Two R-GCNs are used in the MCCA model to encode different knowledge graphs separately, and we investigated the effect of the number of R-GCN layers on the model's performance. Figure 4c shows the results of Recall@10 in the recommendation task on the REDIAL dataset. We noticed that the model performs best when using one layer of R-GCN. This shows that external knowledge can help the model understand user preferences, but as the number of layers increases, it introduces too much noise, leading to a decrease in model performance.

Case studies

In this section, we will use a case to illustrate how MCCA works, as shown in Fig. 5. First, the model extracts the item "*Inception*" in the context, Then, we retrieve the movie's tags and select the top four popular tags, such as "*science fiction action films*", etc., and then get the embeddings of items and tags in the knowledge graph MTKG, and at the same time, we get the embeddings of items in the knowledge graph DBpedia, and then we fuse the embeddings of the items in the two knowledge graphs into a single-item embedding, and the fused embedding is fully mined for features by unsupervised contrastive learning with the embeddings in the two knowledge graphs; For tag embedding, we utilize the unsupervised YAKE method to extract the keywords from the reviews as an assistant for multi-tag fusion to get the final tag embedding. The recommendation module fuses the item embedding and the tag embedding to obtain the user preference representation, predicting movies that the user might be interested in, such as "*Interstellar*". The conversation module utilizes the dialogue context, tags, entities, and the predicted results from the recommendation module to generate responses. If the user dislikes or has already seen the recommended movie, the system will continue to interact with the user and dynamically update the user's preferences. For example, in Fig. 5, the user has already seen "*Interstellar*" and asks for other recom-

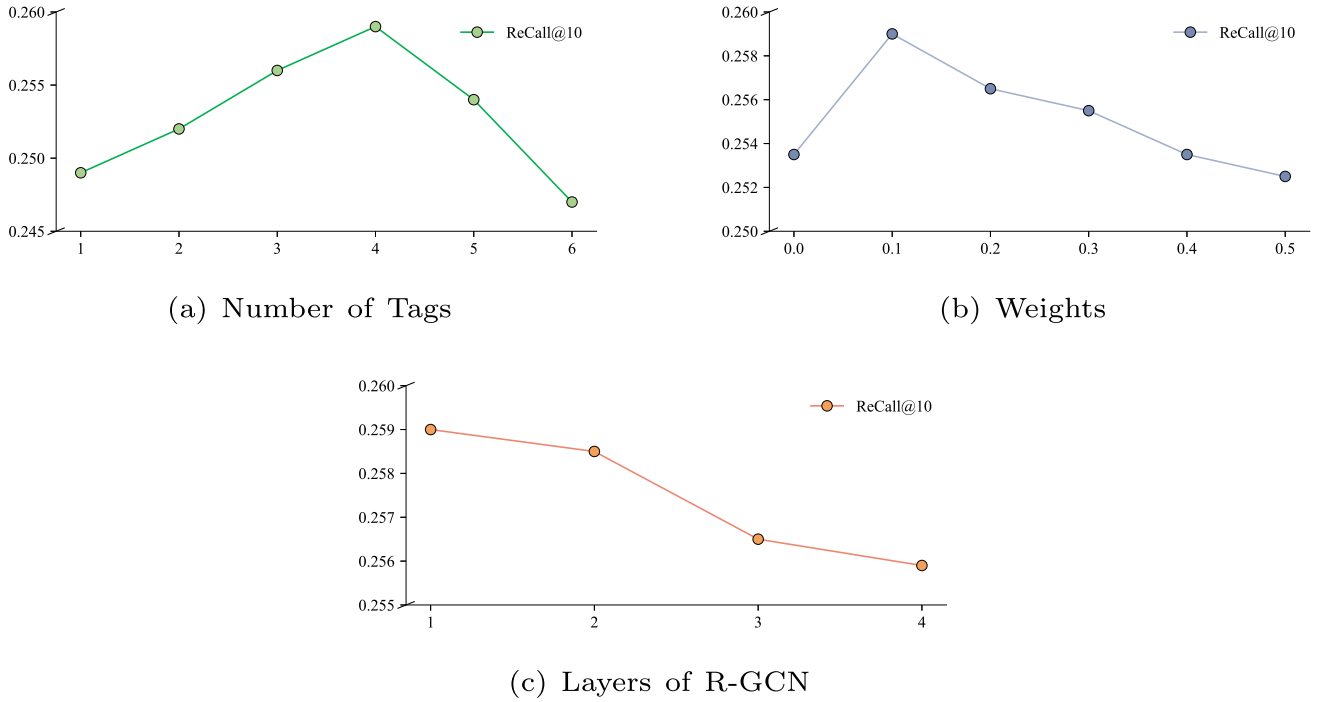

Fig. 4 Parameters analysis on the REDIAL dataset

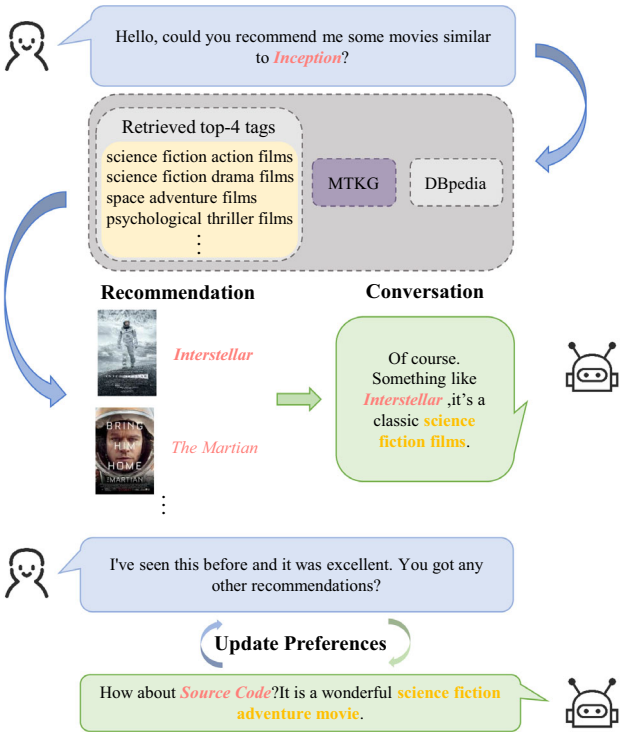

Fig. 5 Case study. Pink font denotes movie items and yellow font denotes descriptive words from tag information

mendations, and the system then makes a re-recommendation based on the dialogue history.

Conclusion

We propose a Multi-source Information Contrastive Learning Collaborative Augmented approach (MCCA) to improve conversational recommender systems, we notice potential internal information that reflects user preferences, i.e., tag information. By constructing a Movie-Tag Knowledge Graph (MTKG) with tag information, we enhance the tag connections between items; Meanwhile, a multi-source item fusion mechanism is designed to bridge the information gap between different knowledge graphs and obtain the fusion item features, based on this, unsupervised contrastive learning is used to optimize the fusion item features, fully exploiting the connections of information in different knowledge graphs; And design a multi-tag fusion mechanism using keywords to assist in enhancing the tag information popularity to get the final tag features. Finally, the acquired tag preferences and item preferences are used to get a better representation of user preferences, and improve the quality of systematic responses. Extensive experiments have shown that our approach can sufficiently bridge the information gap between multi-source information in CRS and fully utilizes information popularity

The extensive introduction of external knowledge in existing studies leads to the noise problem, potentially biasing the captured user preferences. In our future work, we intend to explore more effective methods to address the bias in user preferences caused by external knowledge.

Acknowledgements This study was supported in part by the National Natural Science Foundation of China [Grant No. 62141201], Chongqing University of Technology Graduate Education Quality Development Action Plan Funding Results [Grant No.: gzlcx20232063], Chongqing Banan District Science and Technology Bureau [Grant No.: 2020QC403]. We thank all the anonymous reviewers who generously contributed their time and energy. Their professional advice greatly improved the quality of the manuscript.

Data availability The relevant data in this paper are available from the corresponding author.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Chen Q, Lin J, Zhang Y, Ding M, Cen Y, Yang H, Tang J (2019) Towards knowledge-based recommender dialog system. In: Inui, K., Jiang, J., Ng, V., Wan, X. (eds.) 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. Association for Computational Linguistics, Hong Kong, China . https://doi.org/10.18653/v1/D19-1189
2. Sun Y, Zhang Y (2018) Conversational recommender system. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. SIGIR '18, pp. 235–244. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3209978.3210002
3. Shen Q, Wen H, Tao W, Zhang J, Lv F, Chen Z, Li Z (2022) Deep interest highlight network for click-through rate prediction in trigger-induced recommendation. In: Proceedings of the ACM Web Conference 2022. WWW '22, pp. 422–430. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3485447.3511970
4. Alam M, Iana A, Grote A, Ludwig K, Müller P, Paulheim H (2022) Towards analyzing the bias of news recommender systems using sentiment and stance detection. In: Companion Proceedings of the Web Conference 2022. WWW '22, pp. 448–457. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3487553.3524674
5. Song X, Wu N, Song S, Stojanovic V (2023) Switching-like event-triggered state estimation for reaction-diffusion neural networks against dos attacks. Neural Processing Letters, 8997–9018 https://doi.org/10.1007/s11063-023-11189-1
6. Zhang Z, Song X, Sun X, Stojanovic V (2023) Hybrid-driven-based fuzzy secure filtering for nonlinear parabolic partial differential equation systems with cyber attacks. Int J Adaptive Control Signal Process 37(2):380–398. https://doi.org/10.1002/acs.3529
7. Zhuang Z, Tao H, Chen Y, Stojanovic V, Paszke W (2023) An optimal iterative learning control approach for linear systems with nonuniform trial lengths under input constraints. IEEE Trans Syst Man Cybernet 53(6):3461–3473. https://doi.org/10.1109/TSMC.2022.3225381
8. Tao Y, Tao H, Zhuang Z, Stojanovic V, Paszke W. Quantized iterative learning control of communication-constrained systems with encoding and decoding mechanism. Transactions of the Institute of Measurement and Control https://doi.org/10.1177/01423312231225782
9. Seo Y-D, Kim Y-G, Lee E, Kim H (2021) Group recommender system based on genre preference focusing on reducing the clustering cost. Expert Syst Appl **183**(C) https://doi.org/10.1016/j.eswa.2021.115396
10. Kawai M, Sato H, Shiohama T (2022) Topic model-based recommender systems and their applications to cold-start problems. Expert Syst Appl **202**(C) https://doi.org/10.1016/j.eswa.2022.117129
11. Ma W, Takanobu R, Huang M (2021) CR-walker: Tree-structured graph reasoning and dialog acts for conversational recommendation. In: Moens, M.-F., Huang, X., Specia, L., Yih, S.W.-t. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 1839–1851. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic . https://doi.org/10.18653/v1/2021.emnlp-main.139
12. Zhang C, Huang X, An J (2023) Macr: Multi-information augmented conversational recommender. Expert Systems with Applications 213, 118981 https://doi.org/10.1016/j.eswa.2022.118981
13. Zhou K, Zhao WX, Bian S, Zhou Y, Wen J-R, 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. KDD '20, pp. 1006–1014. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3394486.3403143
14. Lu Y, Bao J, Song Y, Ma Z, Cui S, Wu Y, He X (2021) RevCore: Review-augmented conversational recommendation. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 1161–1173. Association for Computational Linguistics, Online . https://doi.org/10.18653/v1/2021.findings-acl.99
15. Campos R, Mangaravite V, Pasquali A, Jorge A, Nunes C, Jatowt A (2020) Yake! keyword extraction from single documents using multiple local features. Inform Sci 509, 257–289 https://doi.org/10.1016/j.ins.2019.09.013
16. Zhou Y, Zhou K, Zhao WX, Wang C, Jiang P, Hu H (2022) C^2-crs: Coarse-to-fine contrastive learning for conversational recommender system. In: Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. WSDM '22, pp. 1488–1496. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3488560.3498514
17. Ren X, Chen T, Nguyen QVH, Cui L, Huang Z, Yin H (2023) Explicit knowledge graph reasoning for conversational recommendation. ACM Trans Intell Syst Technol. https://doi.org/10.1145/3637216
18. Christakopoulou K, Radlinski F, Hofmann K (2016) Towards conversational recommender systems. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, pp. 815–824. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/2939672.2939746
19. Lei W, Zhang G, He X, Miao Y, Wang X, Chen L, Chua T-S (2020) Interactive path reasoning on graph for conversational recommen-

dation. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. KDD '20, pp. 2073–2083. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3394486.3403258
20. Lei W, He X, Miao Y, Wu Q, Hong R, Kan M-Y, 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. WSDM '20, pp. 304–312. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3336191.3371769
21. Liang Z, Hu H, Xu C, Miao J, He Y, Chen Y, Geng X, Liang F, Jiang D (2021) Learning neural templates for recommender dialogue system. In: Moens, M.-F., Huang, X., Specia, L., Yih, S.W.-t. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 7821–7833. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic . https://doi.org/10.18653/v1/2021.emnlp-main.617
22. Wang L, Hu H, Sha L, Xu C, Jiang D, Wong K-F (2022) RecInDial: A unified framework for conversational recommendation with pretrained language models. In: He, Y., Ji, H., Li, S., Liu, Y., Chang, C.-H. (eds.) Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 489–500. Association for Computational Linguistics, Online only
23. Liu Z, Zhou D, Liu H, Wang H, Niu Z-Y, Wu H, Che W, Liu T, Xiong H (2023) Graph-grounded goal planning for conversational recommendation. IEEE Trans Knowl Data Eng 35(5):4923–4939. https://doi.org/10.1109/TKDE.2022.3147210
24. Sarkar R, Goswami K, Arcan M, McCrae JP (2020) Suggest me a movie for tonight: Leveraging knowledge graphs for conversational recommendation. In: Scott, D., Bel, N., Zong, C. (eds.) Proceedings of the 28th International Conference on Computational Linguistics, pp. 4179–4189. International Committee on Computational Linguistics, Barcelona, Spain (Online) . https://doi.org/10.18653/v1/2020.coling-main.369
25. Zhang T, Liu Y, Li B, Zhong P, Zhang C, Wang H, Miao C (2022) Toward knowledge-enriched conversational recommendation systems. In: Liu, B., Papangelis, A., Ultes, S., Rastogi, A., Chen, Y.-N., Spithourakis, G., Nouri, E., Shi, W. (eds.) Proceedings of the 4th Workshop on NLP for Conversational AI, pp. 212–217. Association for Computational Linguistics, Dublin, Ireland . https://doi.org/10.18653/v1/2022.nlp4convai-1.17
26. Ning Y, Peng J, Liu Q, Huang Y, Sun W, Du Q (2023) Contrastive learning based on category matching for domain adaptation in hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing 61, 1–14 https://doi.org/10.1109/TGRS.2023.3295357
27. Bian S, Zhao WX, Zhou K, Cai J, He Y, Yin C, Wen J-R (2021) Contrastive curriculum learning for sequential user behavior modeling via data augmentation. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. CIKM '21, pp. 3737–3746. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3459637.3481905
28. Zhang L, Sun Z, Zhang J, Wu Y, Xia Y (2023) Conversation-based adaptive relational translation method for next poi recommendation with uncertain check-ins. IEEE Trans Neural Netw Learn Syst 34(10):7810–7823. https://doi.org/10.1109/TNNLS.2022.3146443
29. Lee S, Lee M (2023) Enhancing text comprehension for question answering with contrastive learning. In: Can, B., Mozes, M., Cahyawijaya, S., Saphra, N., Kassner, N., Ravfogel, S., Ravichander, A., Zhao, C., Augenstein, I., Rogers, A., Cho, K., Grefenstette, E., Voita, L. (eds.) Proceedings of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023), pp. 75–86. Association for Computational Linguistics, Toronto, Canada . https://doi.org/10.18653/v1/2023.repl4nlp-1.7
30. Nan G, Qiao R, Xiao Y, Liu J, Leng S, Zhang H, Lu W (2021) Interventional video grounding with dual contrastive learning. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2764–2774 . https://doi.org/10.1109/CVPR46437.2021.00279
31. Zhang H, Koh J, Baldridge J, Lee H, Yang Y (2021) Cross-modal contrastive learning for text-to-image generation. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 833–842. IEEE Computer Society, Los Alamitos, CA, USA . https://doi.org/10.1109/CVPR46437.2021.00089
32. Schlichtkrull M, Kipf TN, Bloem P, Berg R, Titov I, Welling M (2018) Modeling relational data with graph convolutional networks. In: Gangemi, A., Navigli, R., Vidal, M.-E., Hitzler, P., Troncy, R., Hollink, L., Tordai, A., Alam, M. (eds.) The Semantic Web, pp. 593–607. Springer, Cham . https://doi.org/10.1007/978-3-319-93417-4_38
33. van den Oord A, Li Y, Vinyals O (2018) Representation Learning with Contrastive Predictive Coding. arXiv e-prints, 1807–03748 https://doi.org/10.48550/arXiv.1807.03748
34. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. NIPS'17, pp. 6000–6010. Curran Associates Inc., Red Hook, NY, USA
35. Hu X, Mi H, Wen Z, Wang Y, Su Y, Zheng J, Melo G (2021) R2D2: Recursive transformer based on differentiable tree for interpretable hierarchical language modeling. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 4897–4908. Association for Computational Linguistics, Online . https://doi.org/10.18653/v1/2021.acl-long.379
36. Ji H, Huang M (2021) DiscoDVT: Generating long text with discourse-aware discrete variational transformer. In: Moens, M.-F., Huang, X., Specia, L., Yih, S.W.-t. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 4208–4224. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic . https://doi.org/10.18653/v1/2021.emnlp-main.347
37. Karimi Mahabadi R, Ruder S, Dehghani M, Henderson J (2021) Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 565–576. Association for Computational Linguistics, Online . https://doi.org/10.18653/v1/2021.acl-long.47
38. Li J, Song H, Li J (2022) Transformer-based question text generation in the learning system. ICIAI '22, pp. 50–56. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3529466.3529484
39. Gu J, Lu Z, Li H, Li VOK (2016) Incorporating copying mechanism in sequence-to-sequence learning. In: Erk, K., Smith, N.A. (eds.) Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1631–1640. Association for Computational Linguistics, Berlin, Germany . https://doi.org/10.18653/v1/P16-1154
40. Li R, Kahou S, Schulz H, Michalski V, Charlin L, Pal C (2018) Towards deep conversational recommendations. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. NIPS'18, pp. 9748–9758. Curran Associates Inc., Red Hook, NY, USA

41. Hayati SA, Kang D, Zhu Q, Shi W, Yu Z (2020) INSPIRED: Toward sociable recommendation dialog systems. In: Webber, B., Cohn, T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 8142–8152. Association for Computational Linguistics, Online . https://doi.org/10.18653/v1/2020.emnlp-main.654
42. Zhao Z, Zhou K, Wang X, Zhao WX, Pan F, Cao Z, Wen J-R (2023) Alleviating the long-tail problem in conversational recommender systems. In: Proceedings of the 17th ACM Conference on Recommender Systems. RecSys '23, pp. 374–385. Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3604915.3608812
43. Jelinek F, Mercer RL, Bahl LR, Baker JK (2005) Perplexity-a measure of the difficulty of speech recognition tasks. J Acoust Soc Am 62(S1):63–63. https://doi.org/10.1121/1.2016299
44. Li J, Galley M, Brockett C, Gao J, Dolan B (2016) A diversity-promoting objective function for neural conversation models. In: Knight, K., Nenkova, A., Rambow, O. (eds.) Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 110–119. Association for Computational Linguistics, San Diego, California . https://doi.org/10.18653/v1/N16-1014
45. Kingma DP, Ba J (2015) Adam: A method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings