HyperCRS: Hypergraph-Aware Multi-Grained Preference Learning to Burst Filter Bubbles in Conversational Recommendation System

Yongsen Zheng[1,2], **Mingjie Qian**[3], **Guohua Wang**[4†], **Yang Liu**[3],
Ziliang Chen[5], **Mingzhi Mao**[3], **Liang Lin**[3,5], **Kwok-Yan Lam**[1,2†]
[1]Nanyang Technological University, Singapore [2]Digital Trust Centre Singapore
[3]Sun Yat-sen University [4]South China Agricultural University [5]Peng Cheng Laboratory
{yongsen.zheng, kwokyan.lam}@ntu.edu.sg, qianmj7@mail2.sysu.edu.cn, wangguohua@scau.edu.cn
{liuy856, mcsmmz}@mail.sysu.edu.cn, c.ziliang@yahoo.com, linliang@ieee.org

Abstract

The filter bubble is a notorious issue in Recommender Systems (RSs), characterized by users being confined to a limited corpus of information or content that strengthens and amplifies their pre-established preferences and beliefs. Most existing methods primarily aim to analyze filter bubbles in the relatively static recommendation environment. Nevertheless, the filter bubble phenomenon continues to exacerbate as users interact with the system over time. To address these issues, we propose a novel paradigm, Hypergraph-Aware Multi-Grained Preference Learning to Burst Filter Bubbles in Conversational Recommendation System (HyperCRS), aiming to burst filter bubbles by learning multi-grained user preferences during the dynamic user-system interactions via natural language conversations. HyperCRS develops Multi-Grained Hypergraph (user-, item-, and attribute-grained) to explore diverse relations and capture high-order connectivity. It employs Hypergraph-Empowered Policy Learning, which includes Multi-Grained Preference Modeling to model user preferences and Preference-based Decision Making to disrupt filter bubbles during user interactions. Extensive results on four publicly CRS-based datasets show that HyperCRS achieves new state-of-the-art performance, and the superior of bursting filter bubbles in the CRS. Our code is available at https://github.com/zysensmile/HyberCRS.

1 Introduction

Conversational Recommendation Systems (CRSs) (Zheng et al., 2024c,d; Qian et al., 2023) are powerful tools that enable the provision of personalized recommendations to assist users in finding desirable items, which have been widely applied in diverse domains such as e-commerce (Liu et al., 2023), music recommendation (Jin et al., 2023b).

†Corresponding author.

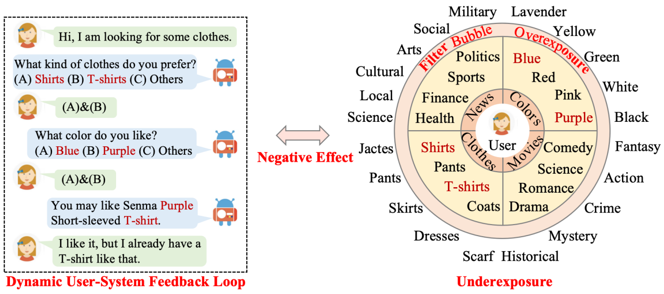

Figure 1: Illustration of the dynamic user-system feedback loop and filter bubble. The filter bubble will be continually exacerbated when the user chats with the system over time.

Nevertheless, CRSs usually face the challenge of filter bubbles, which refer to the phenomenon where users are overexposed to a limited range of information and suggestions that align with their dominant preferences during user-system interactions (Gao et al., 2022). Worse still, the filter bubble issue will be further increasingly intensified as users interact with the system over time. Thus, it is crucial to mitigate filter bubbles in dynamic user-system feedback loop.

Recently, most methods (Li et al., 2023; Ribeiro et al., 2020; Hussein et al., 2020; Liu et al., 2021; Nguyen et al., 2014; Wang et al., 2022; Gao et al., 2022) have primarily focused on filter bubbles in static recommendation settings, often neglecting the negative impacts of dynamic user-system feedback loops. These studies identify core factors contributing to filter bubbles through extensive experiments on large-scale Recommender Systems (RSs). Two key factors have emerged: first, RS learning mechanisms exacerbate filter bubbles by reinforcing users' prevailing tendencies; second, users with limited preference diversity are more likely to become trapped in filter bubbles. While these studies offer valuable insights, they do not effectively address the negative effects of dynamic interactions. Some recent research (Gao et al., 2022)

Findings of the Association for Computational Linguistics: ACL 2025, pages 5597–5608
July 27 - August 1, 2025

has attempted to mitigate filter bubbles in Interactive Recommender Systems (IRSs) by considering user-system interactions, but these methods fail to allow users to express their thoughts through natural language conversations.

Despite their effectiveness, most existing methods have two major limitations: *1) Interaction Manner*. Most methods primarily focus on mitigating filter bubbles for offline recommendations in a static manner, overlooking the negative impact of the dynamic user-system feedback loop, as shown in Fig.1. In reality, filter bubbles often intensify as users interact with the system over time (Steck, 2018; Zheng et al., 2024a). While some studies (Gao et al., 2022; Li et al., 2023; Wang et al., 2022) begin to address this issue in the interactive setting, they still rely on the rigid manners, *e.g.*, click or skip, like or dislike, limiting users' ability to express their diverse and complex preferences through natural language. *2) Preference Exploration*. Many previous researches (Hussein et al., 2020; Liu et al., 2021; Nguyen et al., 2014; Ribeiro et al., 2020) show that modeling diverse user preferences is key to breaking filter bubbles. Thus, they leverage additional knowledge sources to capture a broad spectrum of preferences. Knowledge Graphs (KGs) (Zhou et al., 2020d,a; Liao et al., 2018; Chen et al., 2019; Lin et al., 2020) are commonly used for their rich relationships. However, KGs are limited to simple pairwise connections, which restrict preference learning and complicate the capture of higher-order relationships. Thus, leveraging more powerful graph structures (*e.g.*, hypergraphs) is essential for effectively learning diverse user preferences and breaking filter bubbles.

To address these issues, we proposed a novel end-to-end framework, **Hyper**graph-Aware Multi-Grained Preference Learning to Burst Filter Bubbles in **C**onversational **R**ecommendation **S**ystem (**HyperCRS**), aiming to learn multi-grained user preferences for bursting filter bubbles when users interact with the system over time via natural language conversations. Considering the constraint of KG edges being limited to connecting two nodes, HyperCRS first builds Multi-Grained Hypergraph (*i.e.*, user-, item-, and attribute-grained hypergraph) to explore varying types of multiplex relations closely related to user preferences and capture high-order connectivity information. Afterwards, HyperCRS devises Hypergraph-Aware Policy Learning, which incorporates Multi-Grained Preference Modeling to effectively capture diverse user preferences based on multi-grained hypergraphs, alongside Hypergraph-based Decision Making to automatically mitigate filter bubbles in the CRS by utilizing the learned diverse preferences when the user interacts with the system over time continuously. Lastly, HyperCRS aims to prevent users from becoming trapped in personalized information bubbles, promoting a more diverse and inclusive recommendation experience. To validate its effectiveness, we conduct empirical evaluations of our proposed HyperCRS on four widely-adopted CRS-based benchmarks. The results demonstrate that HyperCRS achieves new state-of-the-art performance, highlighting its superiority in breaking filter bubbles within the CRS.

Overall, our main contributions are included:

- To the best of our knowledge, this is the first work to break filter bubbles in the attributed-based multi-round CRS by learning diverse preferences based on a series of constructed hypergraphs.
- We proposed an end-to-end framework, HyperCRS, aiming to construct multi-grained hypergraphs to capture high-order user relation patterns for exploring diverse preferences, thereby bursting filter bubbles in the CRS.
- Extensive experiments on four publicly widely-adopted CRS-based datasets show that HyperCRS achieves new state-of-the-art performance and its superiority in bursting filter bubbles.

2 Related Work

2.1 Conversational Recommendation System

Conversational Recommendation Systems (CRSs) (Zheng et al., 2024a,b) leverage natural language conversations to suggest items to users, creating a more engaging and interactive experience. These systems can be broadly categorized into two types: attribute-based CRSs (Christakopoulou et al., 2016a; Sun and Zhang, 2018; Zhou et al., 2020b; Zhang et al., 2022a) and generation-based CRSs (Zhou et al., 2022, 2020c; Hayati et al., 2020; Li et al., 2018; Zheng et al., 2024a,b). This work specifically focuses on attribute-based CRSs, which gather various attributes and preferences from users to refine and personalize recommendations. These systems can operate in two distinct modes: single-round (Christakopoulou et al., 2018; Sun and Zhang, 2018), where recommendations are provided in a single interaction, and multi-round (Lei et al., 2020a,b; Xu et al., 2021; Ren et al., 2021; Deng et al., 2021; Tu et al., 2022; Hu et al., 2022; Zhang et al., 2022b), which involve multiple interactions to gather more nuanced information until the task is complete or the session concludes. While multi-round attribute-based CRSs offer a more realistic and dynamic approach to understanding user needs, they often face challenges such as filter bubble issues. Filter bubbles occur when users are repeatedly exposed to a narrow range of recommendations, limiting their exposure to diverse options and potentially skewing their preferences. Our work adopts the multi-round approach to effectively mitigate these filter bubble effects by actively learning and adapting to diverse user preferences throughout the conversation. By fostering a richer dialogue and incorporating varied attributes, we aim to enhance the overall recommendation quality and user satisfaction.

2.2 Filter Bubbles in Recommendation

In the context of recommendation system (Zheng et al., 2021, 2023b,a), the filter bubble (Zheng et al., 2024a) refers to individuals encountering personalized online content that aligns with their beliefs while being shielded from diverse perspectives (Liu et al., 2021; Nguyen et al., 2014). This is driven by algorithmic filtering that prioritizes content based on past behavior, social connections, and demographics. Recent research has identified key factors contributing to filter bubbles. One factor is that individuals with narrower preferences are more susceptible to these bubbles (Ribeiro et al., 2020; Spinelli and Crovella, 2020). Another is that the learning process reinforces a user's existing preferences. Additionally, there is an assumption that user satisfaction correlates with preference, leading to the belief that an overload of favored items won't harm satisfaction (Gao et al., 2022; Li et al., 2023; Wang et al., 2022). However, existing methods primarily address filter bubbles in static environments. In contrast, our HyperCRS aims to disrupt filter bubbles during dynamic user-system interactions through natural language conversations.

3 HyperCRS

Filter bubbles is a notorious issue in the CRS, and it will be increasingly intensified over time as users interact with the system. To address these issues, we propose a novel paradigm, HyperCRS, which consists of Multi-Grained Hypergraph Construction and Hypergraph-Aware Policy Learning. The overall pipeline of HyperCRS is depicted in Fig.2.

3.1 Definition and Preliminaries

RL-based CRS. Let $\mathcal{C}$ represent items, attributes, and types. Each item $v \in \mathcal{V}$ has attributes $\mathcal{P}_v \in \mathcal{P}$, with each attribute p linked to a type $c_p \in \mathcal{C}$. Users u start by stating their preferred attributes p_0, prompting the system to ask for more attributes or recommend items. Key notations include: 1) $\mathcal{P}_{acc}^{(t)}$, $\mathcal{P}_{rej}^{(t)}$, $\mathcal{P}_{cand}^{(t)}$: accepted, rejected, and candidate attributes; 2) $\mathcal{V}_{rej}^{(t)}$, $\mathcal{V}_{cand}^{(t)}$: rejected and candidate items; 3) $\mathcal{F}_u^{(t)}$: friends from interaction history. We use Reinforcement Learning (RL) to simulate the CRS, defining the state and action space as: 1) State: $s_t = [\mathcal{P}_{acc}^{(t)}, \mathcal{P}_{rej}^{(t)}, \mathcal{V}_{rej}^{(t)}, \mathcal{P}_{cand}^{(t)}, \mathcal{V}_{cand}^{(t)}, \mathcal{F}_u^{(t)}]$; 2) Action: $a_t \in \mathcal{A}_t$ includes candidate items and attributes; 3) Transition: User responses change state s_t to s_{t+1}; 4) Reward: Feedback gives an immediate reward $\mathcal{R}(s_t, a_t)$.

Hypergraph. Hypergraphs capture complex interactions among entities through hyperedges. We define multi-grained hypergraphs as $\mathcal{G}^{(t)} = (\mathcal{G}_{user}^{(t)}, \mathcal{G}_{item}^{(t)}, \mathcal{G}_{attr}^{(t)})$, where each component represents user, item, and attribute hypergraphs, respectively. Each hypergraph is formalized as $\mathcal{G}_*^{(t)} = (\mathcal{N}_*^{(t)}, \mathcal{H}_*^{(t)}, \mathcal{A}_*^{(t)})$, consisting of: 1) A node set $\mathcal{N}_*^{(t)}$; 2) A hyperedge set $\mathcal{H}_*^{(t)} = \mathcal{H}_{*\_like}^{(t)} \cup \mathcal{H}_{*\_dis}^{(t)} \cup \mathcal{H}_{*\_f}^{(t)}$, representing Favor, Disfavor, and Relation hyperedges; 3) An adjacency matrix $\mathcal{A}_*^{(t)}$ of size $|\mathcal{N}_*^{(t)}| \times |\mathcal{H}_*^{(t)}|$, indicating weights between nodes and hyperedges.

3.2 Multi-Grained Hypergraph Construction

The key of bursting filter bubbles is to explore diverse user preferences (Nguyen et al., 2014; Hussein et al., 2020; Liu et al., 2021), and thus we develop the Multi-Grained Hypergraph to explore multiplex relations and capture high-order connectivity information to learn diverse user preferences.

User-Grained Hypergraph. The user-grained hypergraph captures multiplex user relationship patterns by considering users' preferences for items and attributes (favor hyperedge), their dislikes (disfavor hyperedge), and the influence of friends with similar preferences (relation hyperedge). This multi-faceted representation allows for a richer understanding of user interactions and behaviors within a recommendation system. Formally, the

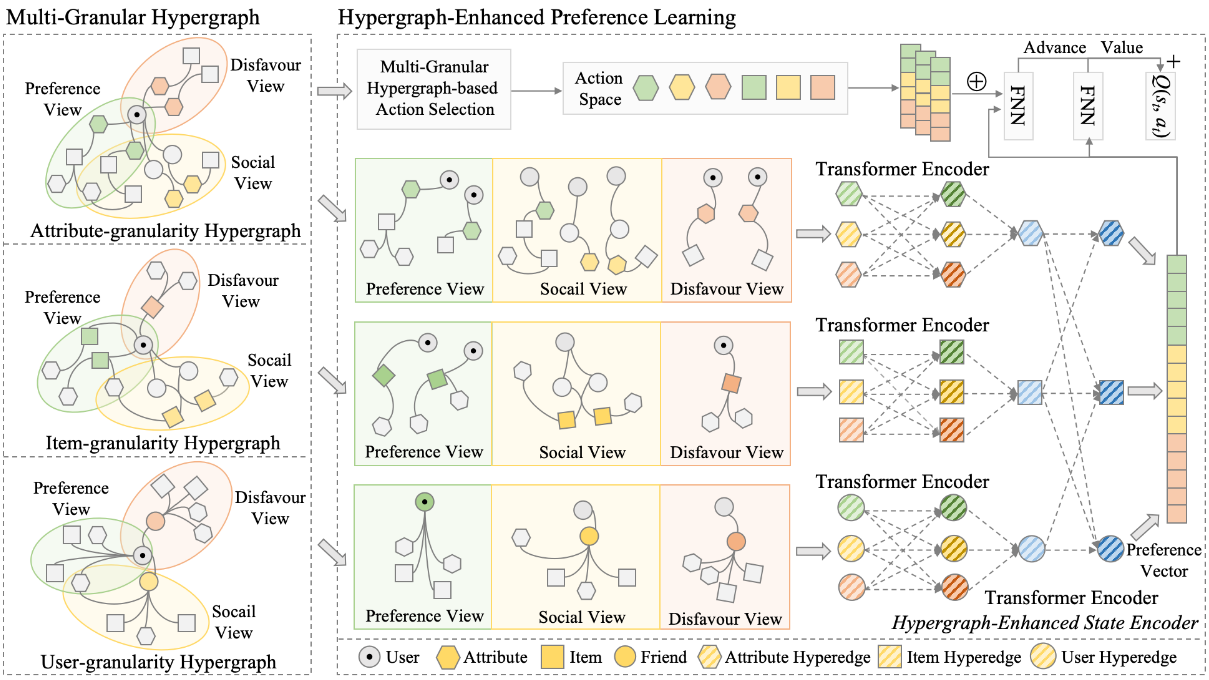

Figure 2: Overview of our HyperCRS, including Multi-Grained Hypergraph and Hypergraph-Aware Policy Learning, aiming to burst filter bubbles by learning diverse user preferences via multi-grained hypergraphs in the CRS.

user-grained hypergraph $\mathcal{G}^{(t)}_{user}$ is defined as:

$$\begin{aligned}
\mathcal{G}^{(t)}_{user} &= (\mathcal{N}^{(t)}_{user}, \mathcal{H}^{(t)}_{user}, \mathcal{A}^{(t)}_{user}); \\
\mathcal{N}^{(t)}_{user} &= \{u\} \cup \mathcal{F}^{(t)}_{u} \cup \mathcal{S}^{(t)}_{u} \cup \mathcal{V}^{(t)}_{u'} \cup \mathcal{P}^{(t)}_{u'}; \\
\mathcal{H}^{(t)}_{user} &= \mathcal{H}^{(t)}_{user_like} \cup \mathcal{H}^{(t)}_{user_dis} \cup \mathcal{H}^{(t)}_{user_f}.
\end{aligned} \tag{1}$$

Here, $\mathcal{S}^{(t)}_{u}$ denotes the strangers to user u, meaning for $s \in \mathcal{S}^{(t)}_{u}$, $\mathcal{V}_u \cap \mathcal{V}_s = \emptyset$. $\mathcal{V}^{(t)}_{u'}$ includes the items interacted with by user $u' \in \{u\} \cup \mathcal{F}^{(t)}_{u} \cup \mathcal{S}^{(t)}_{u}$, while $\mathcal{P}^{(t)}_{u'}$ refers to the attributes that user u' likes. Each hyperedge $h \in \mathcal{H}^{(t)}_{item}$ corresponds to a user u'_h. $\mathcal{H}^{(t)}_{user\_like}$, $\mathcal{H}^{(t)}_{user\_dis}$, and $\mathcal{H}^{(t)}_{user\_f}$ represent user u, strangers to u, and friends of u, respectively. For a given user u, we can derive $\mathcal{A}^{(t)}_{user}$:

$$\mathcal{A}^{(t)}_{i,j} = \begin{cases} 1, & \text{if } h_j \in \mathcal{H}^{(t)}_{user\_*}, n_i = u'_{h_j} \\ \dfrac{1}{|\mathcal{P}^{(t)}_{h_j}|}, & \text{if } h_j \in \mathcal{H}^{(t)}_{user\_*}, n_i \in \mathcal{P}^{(t)}_{h_j} \\ \dfrac{1}{|\mathcal{F}^{(t)}_{h_j}|}, & \text{if } h_j \in \mathcal{H}^{(t)}_{user\_*}, n_i \in \mathcal{F}^{(t)}_{h_j} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Item-Grained Hypergraph. Similarly, the item-grained hypergraph captures multiplex item relationship patterns. It considers users' preferences and dislikes for items, along with the influence of friends who have shown similar preferences based on historical interactions. By incorporating these factors, we effectively capture the connections among items and their relation to user preferences. Formally, it can be expressed as:

$$\begin{aligned}
\mathcal{G}^{(t)}_{item} &= (\mathcal{N}^{(t)}_{item}, \mathcal{H}^{(t)}_{item}, \mathcal{A}^{(t)}_{item}); \\
\mathcal{N}^{(t)}_{item} &= \{u\} \cup \mathcal{F}^{(t)}_{u} \cup \mathcal{V}^{(t)}_{rej} \cup \mathcal{V}^{(t)}_{u} \cup \mathcal{V}^{(t)}_{f} \cup \mathcal{P}^{(t)}_{v}; \\
\mathcal{H}^{(t)}_{item} &= \mathcal{H}^{(t)}_{item_like} \cup \mathcal{H}^{(t)}_{item_dis} \cup \mathcal{H}^{(t)}_{item_f}.
\end{aligned} \tag{3}$$

Here $\mathcal{V}^{(t)}_{u}$, $\mathcal{V}^{(t)}_{f}$, and $\mathcal{P}^{(t)}_{v}$ mean items in $\mathcal{V}^{(t)}_{cand}$ that the user has interacted with, items that user's friends have interacted with, and attributes that belong to the item $v \in \mathcal{V}^{(t)}_{rej} \cup \mathcal{V}^{(t)}_{u} \cup \mathcal{V}^{(t)}_{f}$, respectively. Each hyperedge $h \in \mathcal{H}^{(t)}_{item}$ refers to a item v_h. Given a user u, we can get the $\mathcal{A}^{(t)}_{item}$:

$$\mathcal{A}^{(t)}_{i,j} = \begin{cases} 1, & \text{if } h_j \in \mathcal{H}^{(t)}_{item\_*}, n_i = v_{h_j} \\ \dfrac{1}{|\mathcal{P}^{(t)}_{h_j}|}, & \text{if } h_j \in \mathcal{H}^{(t)}_{item\_*}, n_i \in \mathcal{P}^{(t)}_{h_j} \\ 1, & \text{if } h_j \in \mathcal{H}^{(t)}_{item\_like}, n_i = u \\ -1, & \text{if } h_j \in \mathcal{H}^{(t)}_{item\_dis}, n_i = u \\ \dfrac{1}{|\mathcal{F}^{(t)}_{h_j}|}, & \text{if } h_j \in \mathcal{H}^{(t)}_{item\_f}, n_i \in \mathcal{F}^{(t)}_{h_j} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $\mathcal{P}_{h_j}^{(t)}$ are attributes connected to h_j.

Attribute-Grained Hypergraph. Attribute-grained hypergraphs strive to explore multiplex attribute relations. They take into account the user's favor and disfavor towards attributes, along with the preferences of their friends who share similar tastes based on historical interactions. This approach enables a richer understanding of how attributes influence user behavior and interactions within the network. Formally, the attribute-grained hypergraph can be expressed as:

$$\begin{aligned}&\mathcal{G}_{attr}^{(t)} = (\mathcal{N}_{attr}^{(t)}, \mathcal{H}_{attr}^{(t)}, \mathcal{A}_{attr}^{(t)});\\&\mathcal{N}_{attr}^{(t)} = \{u\} \cup \mathcal{F}_u^{(t)} \cup \mathcal{P}_{rej}^{(t)} \cup \mathcal{P}_{acc}^{(t)} \cup \mathcal{P}_f^{(t)} \cup \mathcal{V}_p^{(t)};\\&\mathcal{H}_{attr}^{(t)} = \mathcal{H}_{attr\_like}^{(t)} \cup \mathcal{H}_{attr\_dis}^{(t)} \cup \mathcal{H}_{attr\_f}^{(t)}.\end{aligned} \tag{5}$$

Here $\mathcal{F}_u^{(t)}$ denotes the user's friends, $\mathcal{P}_f^{(t)}$ represents attributes liked by friend $f \in \mathcal{F}_u^{(t)}$, and $\mathcal{V}_p^{(t)}$ includes items matching attributes $p \in \mathcal{P}_{rej}^{(t)} \cup \mathcal{P}_{acc}^{(t)} \cup \mathcal{P}_f^{(t)}$. Each hyperedge $h \in \mathcal{H}_{attr}^{(t)}$ corresponds to an attribute p_h. $\mathcal{H}_{attr\_like}^{(t)}$, $\mathcal{H}_{attr\_dis}^{(t)}$, and $\mathcal{H}_{attr\_f}^{(t)}$ represent attributes liked, disliked, and liked by the user's friends, respectively. For user u, we derive $\mathcal{A}_{attr}^{(t)}$ as:

$$\mathcal{A}_{i,j}^{(t)} = \begin{cases} 1, & \text{if } h_j \in \mathcal{H}_{attr\_*}^{(t)}, n_i = p_{h_j} \\ \dfrac{1}{|\mathcal{V}_{h_j}^{(t)}|}, & \text{if } h_j \in \mathcal{H}_{attr\_*}^{(t)}, n_i \in \mathcal{V}_{h_j}^{(t)} \\ 1, & \text{if } h_j \in \mathcal{H}_{attr\_like}^{(t)}, n_i = u \\ -1, & \text{if } h_j \in \mathcal{H}_{attr\_dis}^{(t)}, n_i = u \\ \dfrac{1}{|\mathcal{F}_{h_j}^{(t)}|}, & \text{if } h_j \in \mathcal{H}_{attr\_f}^{(t)}, n_i \in \mathcal{F}_{h_j}^{(t)} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $\mathcal{V}_{h_j}^{(t)}$, $\mathcal{F}_{h_j}^{(t)}$ denote items connected to hyperedge h_j and user's friends associated with h_j.

3.3 Hypergraph-Aware Policy Learning

To break filter bubbles, we propose Hypergraph-Aware Policy Learning to capture diverse user preferences, integrating Multi-Grained Preference Modeling and Hypergraph-based Decision Making.

Multi-Grained Preference Modeling. The key to breaking filter bubbles lies in diverse user preferences (Hussein et al., 2020; Liu et al., 2021; Nguyen et al., 2014; Spinelli and Crovella, 2020; Li et al., 2023; Gao et al., 2022). To learn multi-grained user preferences, we use a hypergraph neural network (Zhao et al., 2023) to blend structural and connectivity information, enabling hyperedge representation through a hypergraph message passing approach that facilitates information exchange between nodes. This process can be defined as:

$$\boldsymbol{H} = \boldsymbol{D}_h^{-1} \boldsymbol{A}^\top \boldsymbol{E} \boldsymbol{W}_n, \tag{7}$$

where $\boldsymbol{H} \in \mathbb{R}^{|\mathcal{H}^{(t)}| \times d}$, $\boldsymbol{D}_h$, $\boldsymbol{E} \in \mathbb{R}^{|\mathcal{N}^{(t)}| \times d}$, and $\boldsymbol{W}_n \in \mathbb{R}^{d \times d}$ are hyperedge embeddings, the diagonal matrix that encodes the degrees of the hyperedges, the initial embedding of related nodes, and the weight matrix, respectively.

The hypergraph message-passing paradigm captures multiplex relations but struggles with sequential information. As conversations evolve, new hyperedges form from user approvals or rejections, necessitating higher-level interactions for preference understanding. We integrate the Transformer encoder (Vaswani et al., 2017). Consider the l-th Transformer layer:

$$\boldsymbol{H}_*^{'} = \mathrm{MHA}(\boldsymbol{H}_*^{(l)}, \boldsymbol{H}_*^{(l)}, \boldsymbol{H}_*^{(l)}), \tag{8}$$

$$\boldsymbol{H}_*^{(l+1)} = \mathrm{LN}(\mathrm{FFN}(\boldsymbol{H}_*^{'}) + \boldsymbol{H}_*^{(l)}). \tag{9}$$

Here $* \in \{attr, item, user\}$, and $\mathrm{MHA}(\cdot)$, $\mathrm{LN}(\cdot)$, and $\mathrm{FFN}(\cdot)$ mean multi-head attention, layer normalization, and feed-forward network, respectively. Besides, we use learnable label embeddings $\mathbf{L}$ to explicitly distinguish the category information of each hyperedge, obtaining $\boldsymbol{H}_*^{(0)}$:

$$\boldsymbol{h}_*^{(0)} = \boldsymbol{h}_* \oplus \boldsymbol{l}_{\boldsymbol{h}_*}, \tag{10}$$

where $\boldsymbol{h}_*$, $\boldsymbol{l}_{\boldsymbol{h}_*}$ denote hyperedge embeddings, and the hyperedge's label embedding, respectively. $\oplus$ is the concatenation operation. After undergoing L transformer layers, we can aggregate the information by a mean pooling layer to derive the representation $\widetilde{\boldsymbol{H}}_*$ of each hypergraph:

$$\widetilde{\boldsymbol{H}}_* = \mathsf{MeanPool}(\boldsymbol{H}_*^{(L)}), \tag{11}$$

where $\widetilde{\boldsymbol{H}}_* \in \mathbb{R}^{1 \times d}$. Next, we combine the representations of the three hypergraphs by concatenating them together to induce the diverse user preferences $\boldsymbol{p}_t$ as

$$\boldsymbol{p}_t = \mathrm{FFN}(\widetilde{\boldsymbol{H}}_{all}), \boldsymbol{H}_{all} = \widetilde{\boldsymbol{H}}_{user} \oplus \widetilde{\boldsymbol{H}}_{item} \oplus \widetilde{\boldsymbol{H}}_{attr}, \tag{12}$$

here $\widetilde{\boldsymbol{H}}_{user}$, $\widetilde{\boldsymbol{H}}_{item}$, and $\widetilde{\boldsymbol{H}}_{attr}$ is user-, item-, and attribute-grained preference, respectively.

Hypergraph-based Decision Making. Next, we will use $\boldsymbol{p}_t$ to break filter bubbles in the CRS. To simulate the dynamic user-system feedback loop within an RL framework, the first step is to establish the action space $\mathcal{A}_t$. Since the system aims to explicitly capture user preferences for items and attributes, the action space should include both. Following (Qian et al., 2023), we select the top-K_p items based on the recommendation score $w_v^{(t)}$ and the top-K_p attribute instances based on the entropy score $w_p^{(t)}$. Formally, we rank them as:

$$w_v^{(t)} = \sigma\Big(\mathbf{e}_v^\top \mathbf{e}_u + \frac{1}{|\mathcal{V}_f^{(t)}|} \sum_{v' \in \mathcal{V}_f^{(t)}} \mathbf{e}_v^\top \mathbf{e}_{v'} + \sum_{p \in \mathcal{P}_{acc}^{(t)}} \mathbf{e}_v^\top \mathbf{e}_p - \sum_{p \in \mathcal{P}_{rej}^{(t)}} \mathbf{e}_v^\top \mathbf{e}_p\Big), \tag{13}$$

$$w_p^{(t)} = -\mathrm{P}(p)\cdot\log_2(\mathrm{P}(p)) - (1-\mathrm{P}(p))\cdot\log_2(1-\mathrm{P}(p)), \tag{14}$$

$$\mathrm{P}(p) = \frac{|\mathcal{V}_{cand}^{(t)} \cap \mathcal{V}_p^{(t)}|}{|\mathcal{V}_{cand}^{(t)}|}, \tag{15}$$

where $\sigma(\cdot)$ is the sigmoid function, $\mathbf{e}_u$, $\mathbf{e}_v$ and $\mathbf{e}_p$ are embeddings of the user, item and attribute, respectively. The candidate items $\mathcal{V}_{cand}^{(t)}$ and attributes $\mathcal{P}_{cand}^{(t)}$ are updated by:

$$\mathcal{V}_{cand}^{(t)} = \Big\{ v \mid v \in \mathcal{V}_{p_0} - \mathcal{V}_{rej}^{(t)} \& \mathcal{P}_v \cap \mathcal{P}_{acc}^{(t)} \neq \emptyset \ \& \ \mathcal{P}_v \cap \mathcal{P}_{rej}^{(t)} = \emptyset \Big\}, \tag{16}$$

$$\mathcal{P}_{cand}^{(t)} = \mathcal{P}_{\mathcal{V}_{cand}^{(t)}} - \mathcal{P}_{acc}^{(t)} \cup \mathcal{P}_{rej}^{(t)}. \tag{17}$$

Here, $\mathcal{V}_{p_0}$ are items that meet the initial attribute p_0 specified by the user, and $\mathcal{P}_{\mathcal{V}_{cand}^{(t)}}$ denotes attributes associated with at least one candidate item $\mathcal{V}_{cand}^{(t)}$.

After constructing $\mathcal{A}_t$, we utilize the dueling Q-network (DQN) to determine the next action. Concretely, the expected reward Q-value $Q(s_t, a_t)$ related to s_t and a_t is:

$$Q(s_t, a_t) = f_{\theta_V}(\mathbf{p}_t) + f_{\theta_A}(\mathbf{p}_t, a_t), \tag{18}$$

Here $f_{\theta_V}(\cdot)$ and $f_{\theta_A}(\cdot)$ are two multi-layer perceptrons (MLPs). In the RL framework, the Q-value guides action selection, with the agent choosing the action with the highest Q-value. For item actions, the system recommends the top-K items $\widetilde{V}^{(t)}$ from $\mathcal{A}_t$. For attribute actions, it selects K_a attributes from the same category c in $\mathcal{A}_t$. The Q-value follows the Bellman equation (Bellman and Kalaba, 1957; Zhang et al., 2022a):

$$Q^*(s_t, a_t) = \mathbb{E}_{s_{t+1}}[r_t + \gamma \max_{a_{t+1} \in \mathcal{A}_{t+1}} Q^*(s_{t+1}, a_{t+1}], \tag{19}$$

where $Q^*(s_{t+1}, a_{t+1}) = Q^*(s_{t+1}, a_{t+1}|s_t, a_t)$, and r_t is the reward, and γ means discounted factor.

At each turn, the agent, upon receiving reward r_t, transitions from state s_t to the next. If seeking consultation, accepted and rejected user attributes are as $\mathcal{P}_{cur\_acc}^{(t)}$ and $\mathcal{P}_{cur\_rej}^{(t)}$ respectively, updating the state by merging these attributes. If recommending items $\widetilde{\mathcal{V}}^{(t)}$, all rejected by the user, the state updates by adding these to $\mathcal{V}_{rej}^{(t)}$. The session ends once an item is accepted. The agent's experiences are stored in a replay buffer $\mathcal{D}$, which also stores tuples $(s_t, a_t, r_t, s_{t+1}, \mathcal{A}_{t+1})$. The loss function is:

$$\mathcal{L} = \mathbb{E}_{(s_a, a_t, r_t, s_{t+1}, \mathcal{A}_{t+1}) \sim \mathcal{D}}[(y_t - Q(s_t, a_t; \theta_Q, \theta_M))^2], \tag{20}$$

where θ_M is the parameters of the module designed for hypergraph-based representation learning, while $\theta_Q = \{\theta_V, \theta_A\}$ represents the parameters related to the action-value function, and y_t is the target value as:

$$y_t = r_t + \gamma \max_{a_{t+1} \in \mathcal{A}_{t+1}} Q(s_{t+1}, a_{t+1}; \theta_Q, \theta_M). \tag{21}$$

Here we periodically synchronize a target network Q' with the online network to train HyperCRS.

3.4 Model Complexity Analysis

In this study, model complexity primarily centers on hypergraph constructions. Assuming the training process involves $O(n)$ items (where n can be adjusted through the batch size parameter), constructing a single hypergraph requires iterating over all items, resulting in a complexity of $O(n)$. Consequently, the complexity for constructing the three types of hypergraphs is $O(3 \times n) = O(n)$. This indicates that the running speed can achieve near-linear acceleration on large-scale datasets. Furthermore, this efficiency is crucial for real-time applications where rapid data processing is essential. By optimizing the hypergraph construction, we can significantly reduce computational overhead, allowing for more extensive analyses without sacrificing performance. Ultimately, this approach enhances the scalability of our model, making it suitable for diverse applications in complex systems.

Models	Yelp			LastFM			Amazon Book			MovieLens		
	SR@15↑	AT↓	hDCG↑	SR@15↑	AT↓	hDCG↑	SR@15↑	AT↓	hDCG↑	SR@15↑	AT↓	hDCG↑
Abs Greedy	0.195	14.08	0.069	0.539	10.92	0.251	0.214	13.50	0.092	0.752	4.94	0.481
Max Entropy	0.375	12.57	0.139	0.640	9.62	0.288	0.343	12.21	0.125	0.704	6.93	0.448
CRM	0.223	13.83	0.073	0.597	10.60	0.269	0.309	12.47	0.117	0.654	7.86	0.413
EAR	0.263	13.79	0.098	0.612	9.66	0.276	0.354	12.07	0.132	0.714	6.53	0.457
SCPR	0.413	12.45	0.149	0.751	8.52	0.339	0.428	11.50	0.159	0.812	4.03	0.547
UNICORN	0.456	11.33	0.176	0.778	7.17	0.386	0.538	9.68	0.253	0.862	4.07	0.579
MCMIPL	0.462	11.42	0.179	0.848	7.21	0.351	0.556	10.21	0.237	0.864	4.00	0.581
MHCPL	0.824	10.14	0.254	0.980	5.79	0.388	0.786	8.51	0.302	0.968	2.38	0.647
HutCRS	0.528	11.33	0.175	0.900	6.52	0.348	0.638	9.84	0.227	0.902	4.16	0.475
HyperCRS*	**0.874**	**9.35**	**0.277**	**0.990**	**5.71**	**0.392**	**0.832**	**8.02**	**0.323**	**0.976**	**2.07**	**0.676**

Table 1: Performance comparison of different models on the four datasets. hDCG stands for hDCG@(15, 10). Higher SR@t and hDCG indicate better performance, while lower AT suggests higher efficiency.

4 Experiments and Analyses

We conduct experiments on four CRS-based benchmarks to answer the following questions:

- **RQ1:** How does HyperCRS perform compared with state-of-the-art methods?
- **RQ2:** How does HyperCRS perform at different conversation turns?
- **RQ3:** How does HyperCRS burst filter bubbles?
- **RQ4:** How do the different hypergraphs impact the performance?
- **RQ5:** How do parameters affect our HyperCRS?

4.1 Experimental Protocol

Datasets. To fully evaluate HyperCRS, we use four widely-adopted CRS-based datasets: **Yelp** (Zhang et al., 2022b); **LastFM** (Zhang et al., 2022b); **Amazon Book** (Zhang et al., 2022b); **MovieLens**.
Baselines. We compare our HyperCRS with a series of strong baselines, including **AbsGreedy** (Christakopoulou et al., 2016b), **MaxEntropy**, **CRM** (Sun and Zhang, 2018), **EAR** (Lei et al., 2020a), **SCPR** (Lei et al., 2020b), **UNICORN** (Deng et al., 2021), **MCMIPL** (Zhang et al., 2022a), **MHCPL** (Zhao et al., 2023), **HutCRS** (Qian et al., 2023). Following (Qian et al., 2023), we adopt several metrics: the success rate (SR@t), Average Turns (AT), and hDCG@(T, K). Higher SR@t and hDCG indicate better performance, whereas lower AT reflects greater efficiency.

4.2 Overall Performance (RQ1)

Experimental results in Table 1 highlight the superiority of our model over all compared state-of-the-art methods. Key observations include: 1) HyperCRS outperforms all methods across four datasets. This significant improvement can be attributed to three main factors: (a) The multi-grained hypergraph effectively models diverse multiplex relationships tied to user preferences, capturing high-order interactions among entities within hyperedges; (b) Multi-grained preference modeling excels at uncovering underlying graph structures and higher-order sequential information based on the constructed hypergraphs, enabling the learning of diverse user preferences; (c) Hypergraph-based Decision Making focuses on breaking filter bubbles in the CRS by leveraging diverse user preferences throughout the conversation. 2) Generally, graph-based models (*e.g.*, MHCPL, MCMIPL, UNICORN, SCPR) outperform factorization-based models (*e.g.*, EAR, CRM). This is primarily due to their ability to effectively extract user preferences from the rich information and diverse entities present in the graph. Notably, MCMIPL excels among graph-based approaches, surpassing UNICORN and SCPR by incorporating various aspects of user interests. Furthermore, MHCPL outperforms MCMIPL by leveraging hypergraphs to explore interconnected relationships and dynamically learn user preferences. Most importantly, HyperCRS surpasses all existing graph-based methods, thanks to its multi-grained hypergraphs, which capture user preferences at different levels of granularity.

4.3 Comparison at Different Turns (RQ2)

Apart from measuring the performance solely in the final turn, we also present the success rates at dif-

https://grouplens.org/datasets/movielens/

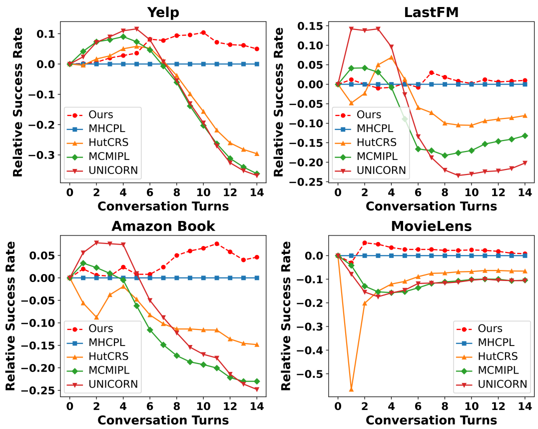

Figure 3: Comparisons at Different Conversation Turns.

Model	Yelp				Yelp			
	A@5↓	A@10↓	G@5↓	G@10↓	L@5↓	L@10↓	D@5↑	D@10↑
UNICORN	0.0020	0.0042	0.6910	0.7692	2.6830	3.6720	0.4067	0.5621
MCMIPL	0.0033	0.0045	0.6856	0.7740	2.7134	3.7013	0.4310	0.5843
MHCPL	0.0030	0.0048	0.7289	0.7450	2.5211	3.7193	0.4839	0.5791
HutCRS	0.0028	0.0039	0.7011	0.7629	2.4567	3.6934	0.4455	0.5939
HyperCRS*	**0.0012**	**0.0023**	**0.6480**	**0.7410**	**2.1034**	**3.4802**	**0.4955**	**0.6011**
Model	LastFM				LastFM			
	A@5↓	A@10↓	G@5↓	G@10↓	L@5↓	L@10↓	D@5↑	D@10↑
UNICORN	0.0038	0.0042	0.5011	0.6411	3.8135	4.8561	0.4914	0.5871
MCMIPL	0.0035	0.0049	0.5123	0.6528	3.9462	4.8732	0.4982	0.5904
MHCPL	0.0037	0.0047	0.5108	0.6255	3.8110	4.7123	0.4914	0.6010
HutCRS	0.0045	0.0046	0.5020	0.6207	3.7855	4.7639	0.4907	0.5951
HyperCRS*	**0.0029**	**0.0035**	**0.4968**	**0.6184**	**3.6554**	**4.6411**	**0.5105**	**0.6040**

Table 2: Study on filter bubbles.

ferent conversation turns, as illustrated in Fig.3. In comparing relative success rates with strong baselines like MHCPL, HutCRS, MCMIP, and UNICORN, we find: 1) HyperCRS dominates all baselines at every turn, mainly because it captures multi-grained user preferences helping negate CRS filter bubbles through effective high-order interactions modeling of entities. 2) In early turns, most methods, including HyperCRS, MHCPL, MCMIPL, fail to surpass baseline due to scant information, prompting preference for questioning over recommending. Conversely, as conversations evolve with more information, these methods adeptly adapt, leading to precision in personalized recommendations. 3) We can also see that UNICORN outperforms other baselines at the beginning turns. Its superior results may be explained by its ability to quickly adapt to user input and context, allowing it to generate relevant and engaging responses right from the start. Furthermore, UNICORN's training on a wide range of conversational data equips it to understand user intent more effectively early on. This combination of contextual awareness and robust training likely accounts for its strong performance in the initial turns.

4.4 Study on Filter Bubbles (RQ3)

Our HyperCRS aims to break filter bubbles in conversational recommendations as users interact with the system over time. Filter bubbles, which arise when algorithms selectively guess what information a user would like to see based on their past behavior, can significantly limit exposure to diverse viewpoints and content. This phenomenon not only narrows the spectrum of information available to users but can also reinforce existing biases, making it crucial to develop systems that encourage broader exploration. To investigate HyperCRS's effectiveness in reducing filter bubbles, we analyze various fairness metrics (Jin et al., 2023a), such as Average Popularity (A@K), Gini Coefficient (G@K), KL-Divergence (L@K), and Difference (D@K). A decrease in A@K, G@K, L@K, and an increase in D@K indicate our proposed HyperCRS can better solve the filter bubble issue.

As shown in Table 2, HyperCRS consistently achieves the lowest values A@K, G@K, L@K and the highest values compared with the strongest baselines. This performance underscores the effectiveness of HyperCRS in mitigating the filter bubble phenomenon, which often restricts users to a narrow range of content based on their previous interactions. For instance, HyperCRS demonstrates remarkable improvements in D@5, achieving enhancements of 179.21%, 130.17%, 100.91%, and 2.34% over prominent models such as UNICORN, MCMIPL, HutCRS, and MHCPL on the Yelp dataset, respectively. These substantial gains indicate that HyperCRS not only introduces a wider variety of recommendations but also successfully diversifies the content presented to users, fostering exploration beyond their established preferences.

4.5 Ablation Studies (RQ4)

To thoroughly evaluate each component of HyperCRS, we conduct a series of ablation studies designed to isolate the impact of specific hypergraph elements on the system's performance. The following configurations are examined: 1) w/o $\mathcal{G}_{user}^{(t)}$: This configuration removes the user-grained hypergraph, which captures the intricate relationships among users based on their interactions and preferences. 2) w/o $\mathcal{G}_{item}^{(t)}$: In this scenario, the item-grained hypergraph is omitted. This hypergraph is essential for understanding the connections and

Model	Yelp				Yelp			
	A@5↓	A@10↓	G@5↓	G@10↓	L@5↓	L@10↓	D@5↑	D@10↑
HyperCRS*	**0.0012**	**0.0023**	**0.6480**	**0.7410**	**2.1034**	**3.4802**	**0.4955**	**0.6011**
w/o $\mathcal{G}_{user}^{(t)}$	0.0021	0.0030	0.6934	0.7645	2.4633	3.5910	0.4822	0.5702
w/o $\mathcal{G}_{item}^{(t)}$	0.0018	0.0032	0.6856	0.7710	2.3942	3.5810	0.4810	0.5783
w/o $\mathcal{G}_{attr}^{(t)}$	0.0033	0.0029	0.6983	0.7681	2.3891	3.5956	0.4805	0.5850
Model	LastFM				LastFM			
	A@5↓	A@10↓	G@5↓	G@10↓	L@5↓	L@10↓	D@5↑	D@10↑
HyperCRS*	**0.0029**	**0.0035**	**0.4968**	**0.6184**	**3.6554**	**4.6411**	**0.5105**	**0.6040**
w/o $\mathcal{G}_{user}^{(t)}$	0.0038	0.0044	0.5013	0.6381	3.8391	4.8561	0.4819	0.5820
w/o $\mathcal{G}_{item}^{(t)}$	0.0040	0.0043	0.5125	0.6490	3.8419	4.8545	0.4900	0.5866
w/o $\mathcal{G}_{attr}^{(t)}$	0.0041	0.0041	0.5066	0.6501	3.8420	4.8601	0.4941	0.5756
Model	Book				Book			
	A@5↓	A@10↓	G@5↓	G@10↓	L@5↓	L@10↓	D@5↑	D@10↑
HyperCRS*	**0.0009**	**0.0013**	**0.3546**	**0.3970**	**2.5046**	**2.9412**	**0.5820**	**0.6490**
w/o $\mathcal{G}_{user}^{(t)}$	0.0015	0.0018	0.3719	0.4012	2.6731	3.0193	0.5718	0.6210
w/o $\mathcal{G}_{item}^{(t)}$	0.0019	0.0016	0.3824	0.4015	2.6810	3.0312	0.5721	0.6241
w/o $\mathcal{G}_{attr}^{(t)}$	0.0014	0.0017	0.3791	0.4045	2.6798	3.0284	0.5720	0.6258
Model	MovieLens				MovieLens			
	A@5↓	A@10↓	G@5↓	G@10↓	L@5↓	L@10↓	D@5↑	D@10↑
HyperCRS*	**0.0018**	**0.0026**	**0.5610**	**0.5823**	**1.6528**	**2.3510**	**0.6870**	**0.7933**
w/o $\mathcal{G}_{user}^{(t)}$	0.0025	0.0031	0.5719	0.5910	1.8012	2.4529	0.6410	0.7305
w/o $\mathcal{G}_{item}^{(t)}$	0.0024	0.0030	0.5730	0.5911	1.8020	2.5655	0.6595	0.7466
w/o $\mathcal{G}_{attr}^{(t)}$	0.0024	0.0035	0.5740	0.5914	1.8045	2.4793	0.6601	0.7511

Table 3: Ablation studies with fairness-aware metrics.

similarities between different items, thereby influencing the diversity of recommendations. 3) w/o $\mathcal{G}_{attr}^{(t)}$: This configuration excludes the attribute-grained hypergraph, which encapsulates the various features and characteristics of items. This component plays a crucial role in enhancing the granularity of recommendations based on specific attributes. The results of these ablation studies are presented in Table 3. Notably, we observe a significant decrease in performance regarding the mitigation of filter bubbles whenever any of the hypergraph components is omitted. This decline underscores the critical importance of each element in the HyperCRS framework.

4.6 Hyperparameter Analysis (RQ5)

We next investigate the impacts of the transformer layer number L on the performance of HyperCRS. As depicted in Figure 4, our analysis reveals several important insights: 1) A suitable number of transformer layers is critical for improving the model's performance. This optimal configuration allows HyperCRS to effectively capture high-order information and complex interactions among various data features. By utilizing an appropriate number of layers, the model can learn intricate patterns and relationships that are essential for generating accurate recommendations. In conversational recommendation systems, where user preferences and contextual nuances play a vital role, the ability to discern these complexities can significantly enhance the relevance and quality of the recommendations provided; 2) Conversely, we find that an excessive number of transformer layers can lead to performance degradation. This decline in performance is likely attributable to overfitting, a phenomenon where the model becomes too tailored to the training data, resulting in a loss of generalization capability. As the number of layers increases, the model's capacity to memorize specific patterns and noise also rises, which can adversely affect its performance on unseen data. This issue is particularly pronounced as the number of neighborhood hops increases, leading to a more complex model that may struggle to maintain robustness across diverse contexts.

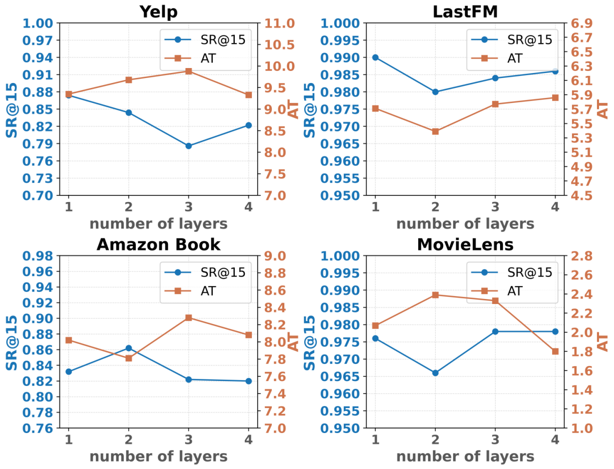

Figure 4: Impact of Layer Number (L).

5 Conclusion

To break filter bubbles in the CRS, we propose HyperCRS, which features Multi-Grained Hypergraph and Hypergraph-Aware Policy Learning. The former models multiplex relations through high-order interactions among entities, while the latter learns diverse preferences via hypergraph structures and sequential information. Extensive experimental results validate the effectiveness of HyperCRS, demonstrating its superiority in breaking filter bubbles within CRS, providing users with a broader spectrum of options and fostering a more enriching online experience. In future work, we plan to integrate Large Language Model (LLM) agents into HyperCRS to enhance conversational capabilities and improve user interactions, and we will also focus on developing user-centric evaluation metrics that assess the effectiveness of LLM interactions within the recommendation framework.

6 Limitations

Despite the exceptional performance of our HyperCRS, several limitations exist. First, it may not fully capture the nuances of individual user behavior, particularly when preferences change rapidly. Additionally, its effectiveness across diverse cultural and demographic contexts remains untested, as user preferences can vary significantly. Lastly, ongoing attention to user privacy and data security is essential, especially when utilizing user-generated content for preference learning.

7 Ethics Statement

The data used in this paper are sourced from open-access repositories, and do not pose any privacy concerns. We are confident that our research adheres to the ethical standards set forth by ACL.

8 Acknowledgements

This research is supported by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Trust Tech Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority.

References

Richard Bellman and Robert Kalaba. 1957. On the role of dynamic programming in statistical communication theory. *IEEE Transactions on Information Theory*, 3(3):197–203.

Qibin Chen, Junyang Lin, Yichang Zhang, Ming Ding, Yukuo Cen, Hongxia Yang, and Jie Tang. 2019. Towards knowledge-based recommender dialog system. *arXiv preprint arXiv:1908.05391*.

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

Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016a. Towards conversational recommender systems. In *International Conference on Knowledge Discovery and Data Mining*, pages 815–824.

Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016b. Towards conversational recommender systems. In *International Conference on Knowledge Discovery and Data Mining*, pages 815–824.

Yang Deng, Yaliang Li, Fei Sun, Bolin Ding, and Wai Lam. 2021. Unified conversational recommendation policy learning via graph-based reinforcement learning. In *Conference on Research and Development in Information Retrieval*, pages 1431–1441.

Chongming Gao, Wenqiang Lei, Jiawei Chen, Shiqi Wang, Xiangnan He, Shijun Li, Biao Li, Yuan Zhang, and Peng Jiang. 2022. CIRS: bursting filter bubbles by counterfactual interactive recommender system. *ACM Transactions on Information Systems*.

Shirley Anugrah Hayati, Dongyeop Kang, Qingxiaoyang Zhu, Weiyan Shi, and Zhou Yu. 2020. INSPIRED: toward sociable recommendation dialog systems. In *Empirical Methods in Natural Language Processing*, pages 8142–8152.

Chenhao Hu, Shuhua Huang, Yansen Zhang, and Yubao Liu. 2022. Learning to infer user implicit preference in conversational recommendation. In *Conference on Research and Development in Information Retrieval*, pages 256–266.

Eslam Hussein, Prerna Juneja, and Tanushree Mitra. 2020. Measuring misinformation in video search platforms: An audit study on youtube. *ACM on Human-Computer Interaction*, 4(CSCW):048:1–048:27.

Di Jin, Luzhi Wang, He Zhang, Yizhen Zheng, Weiping Ding, Feng Xia, and Shirui Pan. 2023a. A survey on fairness-aware recommender systems. *Information Fusion*, 100:101906.

Xin Jin, Wu Zhou, Jinyu Wang, Duo Xu, and Yongsen Zheng. 2023b. An order-complexity aesthetic assessment model for aesthetic-aware music recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6938–6947.

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

Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020b. Interactive path reasoning on graph for conversational recommendation. In *International Conference on Knowledge Discovery and Data Mining*, pages 2073–2083.

Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards deep conversational recommendations. In *Advances in Neural Information Processing Systems*, pages 9748–9758.

Zhenyang Li, Yancheng Dong, Chen Gao, Yizhou Zhao, Dong Li, Jianye Hao, Kai Zhang, Yong Li, and Zhi Wang. 2023. Breaking filter bubble: A reinforcement learning framework of controllable recommender system. In *The Web Conference*, pages 4041–4049. ACM.

Lizi Liao, Yunshan Ma, Xiangnan He, Richang Hong, and Tat-seng Chua. 2018. Knowledge-aware multimodal dialogue systems. In *International Conference on Multimedia*, pages 801–809.

Xiexiong Lin, Weiyu Jian, Jianshan He, Taifeng Wang, and Wei Chu. 2020. Generating informative conversational response using recurrent knowledge-interaction and knowledge-copy. In *Association for Computational Linguistics*, pages 41–52.

Ping Liu, Karthik Shivaram, Aron Culotta, Matthew A. Shapiro, and Mustafa Bilgic. 2021. The interaction between political typology and filter bubbles in news recommendation algorithms. In *The Web Conference*, pages 3791–3801.

Yuanxing Liu, Weinan Zhang, Baohua Dong, Yan Fan, Hang Wang, Fan Feng, Yifan Chen, Ziyu Zhuang, Hengbin Cui, Yongbin Li, and Wanxiang Che. 2023. U-NEED: A fine-grained dataset for user needs-centric e-commerce conversational recommendation. In *Conference on Research and Development in Information Retrieval*, pages 2723–2732. ACM.

Tien T. Nguyen, Pik-Mai Hui, F. Maxwell Harper, Loren G. Terveen, and Joseph A. Konstan. 2014. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *The Web Conference*, pages 677–686. ACM.

Mingjie Qian, Yongsen Zheng, Jinghui Qin, and Liang Lin. 2023. Hutcrs: Hierarchical user-interest tracking for conversational recommender system. In *Conference on Empirical Methods in Natural Language Processing EMNLP*, pages 10281–10290. Association for Computational Linguistics.

Xuhui Ren, Hongzhi Yin, Tong Chen, Hao Wang, Zi Huang, and Kai Zheng. 2021. Learning to ask appropriate questions in conversational recommendation. In *Conference on Research and Development in Information Retrieval*, pages 808–817.

Manoel Horta Ribeiro, Raphael Ottoni, Robert West, Virgílio A. F. Almeida, and Wagner Meira Jr. 2020. Auditing radicalization pathways on youtube. In *Conference on Fairness, Accountability, and Transparency*, pages 131–141. ACM.

Larissa Spinelli and Mark Crovella. 2020. How youtube leads privacy-seeking users away from reliable information. In *Conference on User Modeling, Adaptation, and Personalization*, pages 244–251. ACM.

Harald Steck. 2018. Calibrated recommendations. In *Conference on Recommender Systems*, pages 154–162.

Yueming Sun and Yi Zhang. 2018. Conversational recommender system. In *Conference on Research & Development in Information Retrieval*, pages 235–244.

Quan Tu, Shen Gao, Yanran Li, Jianwei Cui, Bin Wang, and Rui Yan. 2022. Conversational recommendation via hierarchical information modeling. In *Conference on Research and Development in Information Retrieval*, pages 2201–2205.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Wenjie Wang, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. 2022. User-controllable recommendation against filter bubbles. In *Conference on Research and Development in Information Retrieval*, pages 1251–1261. ACM.

Kerui Xu, Jingxuan Yang, Jun Xu, Sheng Gao, Jun Guo, and Ji-Rong Wen. 2021. Adapting user preference to online feedback in conversational recommendation. In *Web Search and Data Mining*, pages 364–372.

Yiming Zhang, Lingfei Wu, Qi Shen, Yitong Pang, Zhihua Wei, Fangli Xu, Bo Long, and Jian Pei. 2022a. Multiple choice questions based multi-interest policy learning for conversational recommendation. In *The Web Conference*, pages 2153–2162.

Yiming Zhang, Lingfei Wu, Qi Shen, Yitong Pang, Zhihua Wei, Fangli Xu, Bo Long, and Jian Pei. 2022b. Multiple choice questions based multi-interest policy learning for conversational recommendation. In *The Web Conference*, pages 2153–2162.

Sen Zhao, Wei Wei, Xian-Ling Mao, Shuai Zhu, Minghui Yang, Zujie Wen, Dangyang Chen, and Feida Zhu. 2023. Multi-view hypergraph contrastive policy learning for conversational recommendation. In *Conference on Research and Development in Information Retrieval SIGIR 2023*, pages 654–664. ACM.

Yongsen Zheng, Ziliang Chen, Jinghui Qin, and Liang Lin. 2024a. Facetcrs: multi-faceted preference learning for pricking filter bubbles in conversational recommender system. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9405–9413.

Yongsen Zheng, Jinghui Qin, Pengxu Wei, Ziliang Chen, and Liang Lin. 2023a. Cipl: counterfactual interactive policy learning to eliminate popularity bias for online recommendation. *IEEE Transactions on Neural Networks and Learning Systems*.

Yongsen Zheng, Guohua Wang, Yang Liu, and Liang Lin. 2024b. Diversity matters: User-centric multi-interest learning for conversational movie recommendation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 9515–9524.

Yongsen Zheng, Pengxu Wei, Ziliang Chen, Yang Cao, and Liang Lin. 2021. Graph-convolved factorization machines for personalized recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(2):1567–1580.

Yongsen Zheng, Pengxu Wei, Ziliang Chen, Chengpei Tang, and Liang Lin. 2023b. Routing user-interest markov tree for scalable personalized knowledge-aware recommendation. *IEEE Transactions on Neural Networks and Learning Systems*.

Yongsen Zheng, Ruilin Xu, Ziliang Chen, Guohua Wang, Mingjie Qian, Jinghui Qin, and Liang Lin. 2024c. Hycorec: Hypergraph-enhanced multi-preference learning for alleviating matthew effect in conversational recommendation. In *Proceedings of Annual Meeting of the Association for Computational Linguistics ACL 2024*, pages 2526–2537. Association for Computational Linguistics.

Yongsen Zheng, Ruilin Xu, Guohua Wang, Liang Lin, and Kwok-Yan Lam. 2024d. Mitigating matthew effect: Multi-hypergraph boosted multi-interest self-supervised learning for conversational recommendation. In *Proceedings of Conference on Empirical Methods in Natural Language Processing, EMNLP 2024*, pages 1455–1466. Association for Computational Linguistics.

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

Kun Zhou, Wayne Xin Zhao, Hui Wang, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020b. Leveraging historical interaction data for improving conversational recommender system. In *Conference on Information and Knowledge Management*, pages 2349–2352.

Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, Xiaoke Wang, and Ji-Rong Wen. 2020c. Towards topic-guided conversational recommender system. In *International Conference on Computational Linguistics*, pages 4128–4139.

Sijin Zhou, Xinyi Dai, Haokun Chen, Weinan Zhang, Kan Ren, Ruiming Tang, Xiuqiang He, and Yong Yu. 2020d. Interactive recommender system via knowledge graph-enhanced reinforcement learning. In *Conference on Research and Development in Information Retrieval*, pages 179–188.

Yuanhang Zhou, Kun Zhou, Wayne Xin Zhao, Cheng Wang, Peng Jiang, and He Hu. 2022. C^2-crs: Coarse-to-fine contrastive learning for conversational recommender system. In *Web Search and Data Mining*, pages 1488–1496. ACM.