A Dual-Channel Heterogeneous Hypergraph Convolutional Network for Dual-target Cross-domain Recommendation

Moyu Zhang⋆ and Zhe Yang⋆(✉)[0000−0003−4980−0343]

School of Computer Science and Technology, Soochow University, China
myzhang23135@stu.suda.edu.cn, yangzhe@suda.edu.cn

Abstract. Cross-domain recommendation (CDR), which aims to alleviate the data sparsity problem in a single domain by integrating complementary data from multiple domains, has become a practical and challenging research direction. Although achieving promising performance, we highlight two problems in existing CDR methods. 1) The representation ability of existing ID-based item embedding is limited. 2) Knowledge transferability across different domains is often insufficient. To solve these problems, we propose a new cross-domain recommendation method, termed Dual-Channel Heterogeneous Hyper Graph Convolutional Network(DHHGCN), which primarily consists of two components: the intra-domain channel layer and the inter-domain channel layer. Concretely, within the intra-domain context, we introduce additional item features and build heterogeneous hypergraphs to model fine-grained high-order correlations, resulting in high-quality user and item representations. In terms of the inter-domain, based on designed similarity matrices, we construct hypergraphs and guide the network to learn the relationships via hypergraph convolution, effectively transferring cross-domain knowledge. Last, an element-wise gating mechanism is designed to integrate domain-specific knowledge with shared cross-domain knowledge, enabling dual-target recommendations. Extensive experiments demonstrate the superiority and effectiveness. Our code is available on GitHub[1].

Keywords: Data Mining · Cross-domain Recommendation · Heterogeneous Hypergraph · Hypergraph Convolutional Network.

1 Introduction

Cross-domain recommendation (CDR) alleviates data sparsity and cold start issues by introducing multi-domain user interactions. Early CDR methods aim to improve performance within a single domain by transferring knowledge from a source domain to a target domain [39, 40]. However, unidirectional methods are prone to accumulating noise in intermediate steps, leading to suboptimal

⋆ Co-first authors with equal contributions

[1] https://github.com/suda-sklcc/DHHGCN

learning. This has led to the development of dual-target CDR, reducing negative transfer from the sparser domain to the richer one [36].

With the development of Graph Neural Networks (GNNs), many methods have been applied to dual-target CDR. Traditional GNN-based CDR methods can be categorized into two types: one constructs separate user-item interaction graphs for each domain, and the other builds a unified interaction graph and trains a shared model to serve all domains [40]. Although promising results are achieved, there are two problems as follows. First, most models [23, 3] only construct graph models based on user-item interactions, neglecting to model the higher-order correlations of useful item attribute information. This limitation prevents capturing the information propagation among multiple heterogeneous nodes. Therefore, they fail to uncover fine-grained higher-order correlations between users and items, considering more information like the price of items. Second, uneven data distribution leads to the problem of negative transfer during cross-domain propagation [37, 38]. Consequently, most models struggle to effectively transfer shared knowledge, resulting in an inability to appropriately balance domain-specific features and shareable features.

Therefore, we highlight two key challenges: (1) How to model heterogeneous attribute information of each domain and obtain fine-grained higher-order correlations to mitigate the sparsity of ID-based single-feature embeddings. (2) How to extract and transmit shareable heterogeneous information across domains to ensure effective and sufficient knowledge transfer. To conquer these challenges, we propose a new cross-domain recommendation method termed Dual-Channel Heterogeneous Hypergraph Convolutional Network (DHHGCN) model. Specifically, to tackle the first challenge, we first select reasonable item attributes as the initial features of the nodes in the graph. Then, we construct heterogeneous hypergraphs and a novel convolution method to learn node embeddings, emphasizing both intra-type and inter-type relationships. Moreover, to address the second challenge, we propose a cross-domain hypergraph construction strategy and an inter-domain convolution method with an element-wise gating mechanism to aggregate embeddings from both intra- and inter-domain channels. The contributions of this paper are summarized as follows:

- We propose DHHGCN, a dual-channel heterogeneous hypergraph convolutional network that integrates heterogeneous attributes for cross-domain recommendation, capturing both domain-specific and shared knowledge.
- We design an inter-domain hypergraph convolution module to transfer shareable information and introduce an element-wise gating mechanism to fuse intra- and inter-domain features for dual-target recommendation.
- Extensive experiments and analyses on real-world datasets demonstrate the superiority and effectiveness of our approaches.

2 Related Work

2.1 Graph Neural Networks based CDR

Traditional recommendation systems [17, 18] have struggled with data sparsity and the cold-start problem, leading to the development of CDR [30]. Conventional methods have difficulty capturing user interest migration and modeling complex item relationships [14, 9]. Graph Neural Networks (GNNs) offer a promising solution by effectively modeling graph-structured data and enabling the exploration of higher-order relationships [4]. BiTGCF [16] utilizes LightGCN [8] to aggregate interaction information and introduces a feature transfer layer to enhance graph encoders. DA-GCN [5] combines Recurrent Neural Networks (RNNs) and GCNs for cross-domain sequential recommendation. Furthermore, some studies have used GNNs to model rich interaction information. ACDN [15] incorporates users' aesthetic features into the CoNet to convey shareable preferences. DDTCDR [13] integrates content information into CDR to address data sparsity. However, most models only capture pairwise correlations, leading to missed insights from higher-order interactions and associations among heterogeneous information. Some models place overly strict dataset requirements when modeling heterogeneous information, limiting their applicability.

2.2 Heterogeneous Graph and Hypergraph

Heterogeneous graphs allow for the modeling of various node and edge types, enabling the encoding of complex relationships among different entities[11, 24, 25]. TrineCDR [38] has proposed effective strategies for leveraging side information in cross-domain setups. Hypergraphs enable the representation of richer relational data via hyperedges, allowing for the capture of higher-order interactions between multiple nodes [21]. Zhou et al. [35] pioneered hypergraph learning and Feng et al. [4] introduced hypergraph convolutional operations to effectively handle complex relationships. Recently, hypergraphs have been applied to recommendation tasks [29], such as the development of the multi-channel hypergraph convolutional network MHCN [28] for social recommendation and the dual-channel hypergraph convolutional network. H^3Trans [26] employs a hierarchical hypergraph network with dynamic item transfer and adaptive user aggregation modules to enhance multi-domain recommendation performance. Furthermore, heterogeneous hypergraphs can effectively represent multiple non-pairwise relationships [22], and as a result, they have gradually been applied in recommendation in recent years. BiPNet [31] and CoHHN [32] proposed a heterogeneous hypergraph network to explore various kinds of information for the session-based recommendation. However, heterogeneous hypergraph has not been fully explored and utilized in cross-domain recommendation.

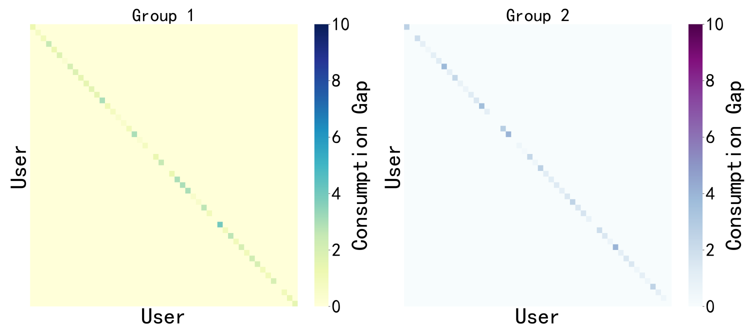

Fig. 1. The heatmap of consumption differences for each user across different domains, with group 1 in Beauty & Health and group 2 in Cloth & Sports. Darker colors indicate larger differences, while lighter colors represent smaller differences.

3 Motivation and Framework

3.1 Motivation Study and Data Processing

We examine the Amazon, representative e-commerce dataset to identify suitable item attributes for feature embedding. We reveal that price is significantly correlated with user preferences. However, absolute price cannot determine whether it is expensive or cheap [34]. Therefore, we categorize absolute prices into multiple intervals based on item categories. The formula of the price level p_i is as follows:

$$p_i = \text{round}\left(\frac{\phi(x_i) - \phi(x_{\min})}{\phi(x_{\max}) - \phi(x_{\min})} \times \rho\right), \tag{1}$$

where round$(*)$ denotes the rounding operation, x_i represents the absolute price of an item, $x_{\min}$ and $x_{\max}$ represent the cheapest and the most expensive in each category respectively, ρ is the total price level. The function $\phi(x)$ denotes the cumulative distribution function of the logistic distribution. Although users exhibit varying preferences across different item categories, they tend to demonstrate consistency in their overall spending levels across various domains. Therefore, to investigate this, we randomly select 100 users with purchase records in different domains. By comparing their average spending levels, we can validate the rationale for incorporating price attributes. Fig. 1 shows that user spending is similar across different domains, despite varying item prices. Comparative analysis reveals that the relationship between item prices and categories is important for CDR. While price ranges vary within the same domain, recommendations based on users' overall spending can help bridge differences across categories and price ranges, providing more personalized recommendations.

3.2 Overview of the Framework

This paper presents a Dual-Channel Heterogeneous Hypergraph Convolutional Network (DHHGCN) for the dual-target CDR. As illustrated in Fig. 2. The

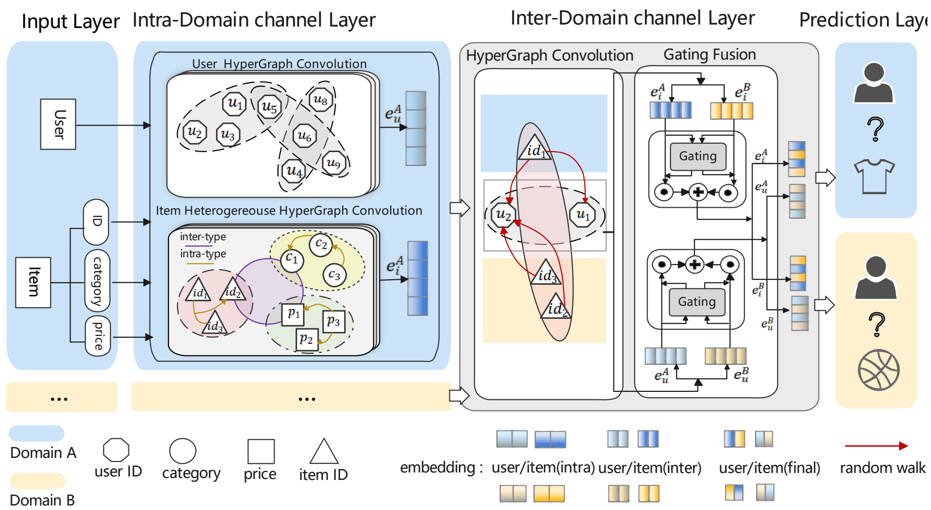

Fig. 2. The framework of DHHGCN. The modeling approach for the intra-domain channel layer is the same across all domains, so for simplicity, we use domain A as an example in the model diagram and omit domain B.

framework consists of four main components: input layer, intra-domain channel layer, inter-domain channel layer, and prediction layer. The input layer takes in heterogeneous information as initial feature embeddings. In the intra-domain channel layer, we construct user hypergraphs and item heterogeneous hypergraphs for each domain, we perform different types of convolution operations to extract fine-grained domain-specific features. In the inter-domain channel layer, we construct inter-domain user and item hypergraphs based on the similarity matrix, respectively, we also implement a cross-domain graph convolution operation and design an element-wise gating mechanism to enhance the flexibility and accuracy of feature fusion. Finally, in the prediction layer, we provide recommendations for both domains based on the obtained node embedding representations.

4 THE PROPOSED METHOD

4.1 Preliminary

Our goal is to enhance recommendation performance across two domains, A and B. The overlapping users between the two domains are denoted as $\mathcal{U} = \{u_1, u_2, u_3, \ldots, u_m\}$, with m users. The items set for each domain are $\mathcal{I}^{\mathrm{A}} = \{\mathrm{I}_1^{\mathrm{A}}, \mathrm{I}_2^{\mathrm{A}}, \mathrm{I}_3^{\mathrm{A}}, \ldots, \mathrm{I}_n^{\mathrm{A}}\}$, $\mathcal{I}^{\mathrm{B}} = \{\mathrm{I}_1^{\mathrm{B}}, \mathrm{I}_2^{\mathrm{B}}, \mathrm{I}_3^{\mathrm{B}}, \ldots, \mathrm{I}_k^{\mathrm{B}}\}$, where n and k represent the number of items in domain A and B, respectively. User-item interactions are represented by matrices $\mathbf{R}^{\mathrm{A}} \in \{0,1\}^{m\times n}$ and $\mathbf{R}^{\mathrm{B}} \in \{0,1\}^{m\times k}$. $\mathbf{H} \in \{0,1\}^{v\times\ e}$represent the association matrix, the adjacency matrix is $a_{ij} \in \{0,1\}^{m\times m}$, $\sigma(*)$ is the activation function. The graph structures are as follows:

Intra-Domain Graph: The user hypergraphs are denoted as $\mathbf{G}_{\mathcal{U}}^{\mathrm{A}}\left(\mathcal{V}_u, \mathcal{E}^{\mathrm{A}}\right)$ and $\mathbf{G}_{\mathcal{U}}^{\mathrm{B}}\left(\mathcal{V}_u, \mathcal{E}^{\mathrm{B}}\right)$. The item heterogeneous hypergraphs, represented as $\mathbf{G}_{\mathcal{I}}^{\mathrm{A}}\left(\mathcal{V}_{\mathrm{A}}^t, \mathcal{E}_{\mathrm{A}}^\tau\right)$ and $\mathbf{G}_{\mathcal{I}}^{\mathrm{B}}\left(\mathcal{V}_{\mathrm{B}}^t, \mathcal{E}_{\mathrm{B}}^\tau\right)$, consist of different node and edge sets $\mathcal{V}^t$ and $\mathcal{E}^\tau$. There are three types of nodes: items ID nodes($\mathcal{V}^{id}$), price nodes($\mathcal{V}^p$) and category nodes($\mathcal{V}^c$); The six types of edges defined as $\mathcal{E}_{vi}$, $\mathcal{E}_{vp}$, $\mathcal{E}_{vc}$ represent item, price and category co-occurrence respectively, $\mathcal{E}_{pv}$ represents price-attribute items, $\mathcal{E}_{pc}$ denotes price-attribute categories, $\mathcal{E}_{cv}$ indicates category-attribute items.

Inter-Domain Graph: $\mathbf{G}_{\mathcal{U}}^{\mathrm{C}}(\mathcal{V}_u, \mathcal{E}_u)$, $\mathbf{G}_{\mathcal{I}}^{\mathrm{C}}(\mathcal{V}_i, \mathcal{E}_i)$ represent the user and item hypergraphs, respectively. The similarity matrices constructed for the user and item hypergraphs are denoted as $\mathbf{H}_{\mathrm{SI}}$ and $\mathbf{H}_{\mathrm{SU}}$. The degree matrices for the vertices and edges are represented by $\mathbf{D}_{\mathcal{V}} \in \mathrm{R}^{v \times v}$, $\mathbf{D}_{\mathcal{E}} \in \mathrm{R}^{e \times e}$.

4.2 The Intra-Domain Channel Layer:

We first construct intra-domain hypergraphs and heterogeneous hypergraphs for users and items based on high-order similarity and co-occurrence relationships. Then, we employ various convolution algorithms to improve information propagation and feature learning. The intra-domain channel is built using domain A as an example, as domains A and B are identical.

Hypergraph Construction. We first define co-occurrence and higher-order relationships. In user-item interaction data, if both u_j and u_k interact with item I_i, they are considered to have a co-occurrence relationship. Based on this, we construct an adjacency matrix $\mathbf{H}_u = \mathbf{R}^{\mathrm{A}}$ for users with co-occurrence relationships. For higher-order relationships [10], we only select second-order correlations [7] to avoid noise. Specifically, if u_k connects u_j and I_i, then u_j is considered a higher-order reachable user of I_i. We extract all such users as the hyperedge $\mathbf{J}_u\left(\mathrm{I}_i\right)$. The user hypergraph $\widetilde{\mathbf{H}}_u$ is computed as follows:

$$\widetilde{\mathbf{H}}_u = \mathbf{H}_u \times \min\left(1, \mathbf{H}_u^{\mathrm{T}} \times \mathbf{H}_u\right). \tag{2}$$

Since hyperedges can contain multiple nodes, an excessive number of hyperedges may cause information redundancy. To address this, we propose a hyperedge sparsification process, retaining only highly similar users or items. The importance of hyperedges is measured by their hyperdegree $\mathbf{D}_u \in \{0,1\}^{n \times n}$, thus we filter and prune based on their hyperdegree, removing hyperedges with lower hyperdegree. We choose the mean degree x of $\mathbf{D}_u$ as the threshold for filtering redundant edges.

$$\left(\mathbf{F}_u\right)_{ii} = \begin{cases} 1, \, if \, \left(\mathbf{D}_u\right)_{ii} > x \\ 0, \, \text{otherwise.} \end{cases} \tag{3}$$

The final user hypergraph representation, derived from the co-occurrence relationship-based association matrix $\mathbf{H}_u$, is shown below, where $||$ denotes the concatenation operation.

$$\mathbf{H}_{\mathcal{U}} = \mathbf{H}_u \| \widetilde{\mathbf{H}}_u \times \mathbf{F}_u. \tag{4}$$

We create a heterogeneous hypergraph for items using co-occurrence relationships. This hypergraph has three types of nodes: item ID, price, and category features, i.e., $\mathcal{V}^t = \mathcal{V}^{id} \cup \mathcal{V}^p \cup \mathcal{V}^c$; We design six types of edges to capture diverse feature relationships, enabling a more comprehensive representation of user-item associations. $\mathcal{E}_{vi}$ models co-occurrence relationships, reflecting users' preferred item sets. $\mathcal{E}_{vp}$ and $\mathcal{E}_{vc}$ capture users' price sensitivity across different price ranges. Further, $\mathcal{E}_{pi}$ and $\mathcal{E}_{pc}$ represent items and categories within the same price range, and $\mathcal{E}_{ci}$ connects items in the same category. The final heterogeneous edge set is $\mathcal{E}^{\tau} = \mathcal{E}_{vi} \cup \mathcal{E}_{vp} \cup \mathcal{E}_{vc} \cup \mathcal{E}_{pi} \cup \mathcal{E}_{pc} \cup \mathcal{E}_{ci}$.

Hypergraph Convolution. Based on the constructed user hypergraph, we adopt traditional hypergraph convolution [27] and incorporate a residual connection mechanism to update node embeddings $\mathbf{E}_{\mathcal{U}}$:

$$\mathbf{E}_{\mathcal{U}}^{(\ell+1)} = \sigma\left(\mathbf{D}_{u_v}^{-1}\mathbf{H}_{\mathcal{U}}\mathbf{D}_{u_e}\mathbf{H}_{\mathcal{U}}^{T}\mathbf{E}_{\mathcal{U}}^{(\ell)}\mathbf{W}^{(\ell)} + \mathbf{E}_{\mathcal{U}}^{(\ell))}\right), \tag{5}$$

where $\mathbf{W}^{(\ell)} \in \mathbb{R}^{c(\ell)\times c(\ell+1)}$ denotes the shared parameters, c(ℓ) denotes the number of convolutional layers.

For item heterogeneous hypergraph convolution, we first group neighboring nodes of the same type to aggregate relevant information for each node type, thereby distinguishing the importance of nodes within a specific type. For any node $\mathbf{v}_i$, we assume $(\mathcal{N}_t)_i$ as the set of neighboring nodes of type t. Considering that nodes of the same type carry homogeneous information, the intra-type convolution aggregates information by performing a weighted summation of the neighboring nodes $(\mathbf{v}_t)_j$, resulting in the node embedding representation $(\mathbf{e}_t)_i$.

$$(\mathbf{e}_t)_i = \sum_j \frac{\exp\left((\mathbf{v}_t)_i^{\top}\mathbf{W}(\mathbf{v}_t)_j\right)}{\sum_{(\mathbf{v}_t)_j\in(\mathcal{N}_t)_i}\exp\left((\mathbf{v}_t)_i^{\top}\mathbf{W}(\mathbf{v}_t)_j\right)}(\mathbf{v}_t)_j, \tag{6}$$

where $\mathbf{W} \in \mathbb{R}^{d\times d}$ is a learnable parameter used to evaluate the similarity between nodes. Thus, the intra-type convolution yields the feature embedding representation $\mathbf{E}_t = f_a(\mathcal{N}_t)$. Next, we designed an inter-type convolution to aggregate relevant heterogeneous information from different types, enriching the feature representation of item nodes from various perspectives. Specifically, we introduce a gating mechanism to perform weighted integration of embeddings from each type, adaptively assigning weights to each type. This allows the model to adjust the information fusion method based on the importance of features from each type. Formally,

$$\mathbf{O}_1 = \sigma\left(\mathbf{E}'_{t_1} + \mathbf{W}_1\mathbf{E}_{t_2}\right), \tag{7}$$

$$\mathbf{O}_2 = \sigma\left(\mathbf{E}'_{t_1} + \mathbf{W}_2\mathbf{E}_{t_3}\right), \tag{8}$$

$$\mathbf{E}'_{t_1} = \mathbf{W}_3\left(\mathbf{E}_{t_1} \| \mathbf{E}_{t_2} \|\mathbf{E}_{t_3}\right), \tag{9}$$

where $\mathbf{W}_{k\in 1,2,3}$ are learnable parameters, t_1, t_2, t_3 are there types of nodes. The formula for updating node embedding based on neighboring nodes is $\widetilde{\mathbf{E}}_t = \mathbf{A}_t\mathbf{E}_t(t \in id, p, c)$. Additionally, a residual structure is applied to prevent gradient explosion. We update three types of nodes through a two-type convolution, the final calculation formulas for their embedding representations are given:

$$\mathbf{E}_p^{(\ell+1)} = \mathbf{E}_p^{(\ell)} + \mathbf{O}_1 \odot \mathbf{E}_{id}^{(\ell)} + \mathbf{O}_2 \odot \mathbf{E}_c^{(\ell)} + \widetilde{\mathbf{E}}_p^{(\ell)}, \tag{10}$$

$$\mathbf{E}_c^{(\ell+1)} = \mathbf{E}_c^{(\ell)} + \mathbf{O}_1 \odot \mathbf{E}_{id}^{(\ell)} + \mathbf{O}_2 \odot \mathbf{E}_p^{(\ell)} + \widetilde{\mathbf{E}}_c^{(\ell)}, \tag{11}$$

$$\mathbf{E}_{id}^{(\ell+1)} = \sigma\left(\mathbf{W}^{(\ell)}\mathbf{E}_{id}^{(\ell)} + \mathbf{E}_{id}^{(\ell)}\right) + \mathbf{O}_1 \odot \mathbf{E}_p^{(\ell)} + \mathbf{O}_2 \odot \mathbf{E}_c^{(\ell)} + \widetilde{\mathbf{E}}_{id}^{(\ell)}, \tag{12}$$

where $\odot$ denotes the element-wise product. In summary, the final embedding representations for domain-specific items and users we obtained are $\mathbf{E}_{\mathcal{U}_\mathrm{A}}$ $\mathbf{E}_{\mathcal{U}_\mathrm{B}}$, $\mathbf{E}_{\mathcal{I}_\mathrm{A}}$, $\mathbf{E}_{\mathcal{I}_\mathrm{B}}$, and $\mathbf{E}_{\mathcal{I}_\mathrm{A}} = \mathbf{E}_{id}^{\mathrm{A}}$, $\mathbf{E}_{\mathcal{I}_\mathrm{B}} = \mathbf{E}_{id}^{\mathrm{B}}$.

4.3 The Inter-Domain Channel Layer:

CDR could use the similarity of user behavior and item attributes across domains. Random walks, by iteratively traversing the graph, can naturally capture high-order relational information. To this end, we propose a random walk-based method to identify similar user/item pairs and construct cross-domain user and item hypergraphs. Taking nodes u and v as an example, we perform multiple random walks starting from these nodes. Two stopping count vectors, $\mathbf{n}_u$ and $\mathbf{n}_v$, record the number of random walks that terminate at each overlapping node. The similarity $\mathbf{s}(u, v)$ is then calculated using cosine similarity as follows:

$$\mathbf{s}(u, v) = \frac{\mathbf{n}_u^T \times \mathbf{n}_v}{\|\mathbf{n}_u\|_2 \times \|\mathbf{n}_v\|_2} = \widehat{\mathbf{n}}_u^T \widehat{\mathbf{n}}_v. \tag{13}$$

After normalization, to further enhance computational efficiency, we convert the similarity matrix into a discrete similarity level matrix. This maps continuous similarity values into discrete levels, enabling the model to dynamically adjust node similarity weights. We employ two thresholds to minimize noise, capture long-tail information, and enhance correlations between highly similar nodes. This approach supports fine-grained feature learning and information propagation in the graph convolutional network. Taking items in domain A as an example, the final similarity matrix is as follows:

$$\mathbf{H}_{\mathrm{SI}}^{\mathrm{A}} = \begin{cases} 0, \, if \;\, \mathbf{s}(u, v) \leq t_1, \\ 1, \, if \;\, t_1 < \mathbf{s}(u, v) < t_2, \\ 2, \, if \;\, \mathbf{s}(u, v) \geq t_2. \end{cases} \tag{14}$$

Since the inter-domain similarity matrix connects nodes across both domains, $\mathbf{H}_{\mathrm{SI}}^{\mathrm{A}} = \mathbf{H}_{\mathrm{SI}} = \left(\mathbf{H}_{\mathrm{SI}}^{\mathrm{B}}\right)^{\mathrm{T}}$ indicates that the similarity matrices between domains A and B are symmetric. We then apply an adaptive hypergraph convolutional neural network on user hypergraph and item hypergraph, respectively:

$$\mathbf{P}_{\mathcal{I}_\mathrm{A}}^{(\ell)} = \sigma\left(\mathbf{D}_{\mathrm{H}_{\mathrm{SI}_e}^{\mathrm{A}}}^{-1/2} \mathbf{H}_{\mathrm{SI}}^{\mathrm{A}} \mathbf{D}_{\mathrm{H}_{\mathrm{SI}_v}^{\mathrm{A}}}^{-1/2} \mathbf{E}_{\mathcal{I}_\mathrm{B}}^{(\ell)}\right). \tag{15}$$

The resulting embeddings $\mathbf{P}_{\mathcal{I}_{\mathrm{A}}}$ represent the inter-domain item embeddings. Finally, we design an element-wise gating mechanism to integrate feature representations of users and items learned from intra and inter-channels. This mechanism assigns an adaptive weight to each feature element, enabling flexible and efficient feature aggregation. Formally,

$$\widetilde{\mathbf{E}}_{\mathcal{I}_{\mathrm{A}}}^{(\ell+1)} = \mathbf{G}_{\mathcal{I}}^{\mathrm{A}} \odot \mathbf{E}_{\mathcal{I}_{\mathrm{A}}}^{(\ell)} + \left(1 - \mathbf{G}_{\mathcal{I}}^{\mathrm{A}}\right) \odot \mathbf{P}_{\mathcal{I}_{\mathrm{A}}}^{(\ell)}, \tag{16}$$

$$\mathbf{G}_{\mathcal{I}}^{\mathrm{A}} = \sigma\left(\mathbf{W}_1 \mathbf{E}_{\mathcal{I}_{\mathrm{A}}}^{(\ell)} + \mathbf{W}_2 \mathbf{P}_{\mathcal{I}_{\mathrm{A}}}^{(\ell)}\right), \tag{17}$$

$$\widetilde{\mathbf{E}}_{\mathcal{I}_{\mathrm{A}}} = \widetilde{\mathbf{E}}_{\mathcal{I}_{\mathrm{A}}}^{1} \| \widetilde{\mathbf{E}}_{\mathcal{I}_{\mathrm{A}}}^{2} \| \ldots \| \widetilde{\mathbf{E}}_{\mathcal{I}_{\mathrm{A}}}^{\ell}, \tag{18}$$

where ℓ represents the number of neural network layers, $\mathbf{G}_{\mathcal{I}}^{\mathrm{A}}$ is the element-wise gating weight vector for item features, and $\mathbf{W}_1$ and $\mathbf{W}_2$ are weight matrices in the gating mechanism, used to map item features into the gating space.

4.4 Prediction Layer:

Based on the final embedding representations of users and items obtained, we utilize cosine similarity to calculate the likelihood $\widetilde{\mathbf{R}}_{ij}$ of user-item interactions within each domain. Taking domain A as an example:

$$\widetilde{\mathbf{R}}_{ij}^{\mathrm{A}} = \frac{(\widetilde{\mathbf{E}}_{\mathcal{U}_{\mathrm{A}}})_i \times (\widetilde{\mathbf{E}}_{\mathcal{I}_{\mathrm{A}}})_j}{\| (\widetilde{\mathbf{E}}_{\mathcal{U}_{\mathrm{A}}})_i \| \, (\widetilde{\mathbf{E}}_{\mathcal{I}_{\mathrm{A}}})_j \|} \tag{19}$$

We select binary cross-entropy loss to optimize the model. The formula for calculating the loss function is:

$$\mathcal{L}_{\mathrm{A}} = \sum_{\mathcal{U}_i^{\mathrm{A}} \in \mathcal{U}_{\mathrm{A}}, \mathcal{I}_j^{\mathrm{A}} \in \mathcal{I}_{\mathrm{A}}} \mathbf{R}_{ij}^{\mathrm{A}} \log \widehat{\mathbf{R}}_{ij}^{\mathrm{A}} + (1 - \mathbf{R}_{ij}^{\mathrm{A}}) \log(1 - \widehat{\mathbf{R}}_{ij}^{\mathrm{A}}). \tag{20}$$

Since the objective of our model is to simultaneously enhance the recommendation performance in both domains, the overall loss function is composed of the loss from domain A and the loss from domain B, as follows: $\mathcal{L} = \mathcal{L}_A + \mathcal{L}_B$.

5 Experimental Settings

Datasets. This paper focuses on e-commerce platforms. Therefore, we utilize datasets from real-world e-commerce platform[2], which is widely used in CDR model experiments and contains rich item attributes [1, 6, 16]. We construct pairwise combinations of the four datasets and identify shared users between each domain pair for CDR scenario. We preprocess the data by treating ratings over 2 as positive samples and considering the rest as negative. We also filter out items with fewer than 5 total interactions.

[2] http://jmcauley.ucsd.edu/data/amazon/

Evaluation Metrics. We employ the Leave-One-Out method [2, 7] for evaluation. To assess the model's performance, we select three widely used metrics [2, 7], namely Hit Rate (HR), Normalized Discounted Cumulative Gain (NDCG), and Mean Reciprocal Rank (MRR). The prediction ranking cutoff is set to $\text{topK} = 5, 10$.

Parameter Settings. We design key training strategies to improve model effectiveness and stability. The batch size is 100, the learning rate is 0.001, and the number of epochs is 200. The model is a hypergraph convolutional neural network with two layers and an embedding vector size of 128. The inter-domain similarity matrix threshold values are 2 and 4. Additionally, we use the Adam optimizer [12] to train the model and implement an early stopping mechanism to prevent overfitting.

Comparison Methods. We select nine models for comparison with our model, covering both single-domain and cross-domain recommendation approaches:

SDR: LightGCN [8] simplifies recommendation using graph networks. CoHHN [32] incorporates item attributes via hypergraph dual-channel aggregation. BiPNet [31] introduces a dual-preference heterogeneous hypergraph network that captures user price and interest preferences;

CDR: PPGN [33] integrates interaction information from multiple domains into a joint graph and shares user features. DisenCDR [1] enhances CDR performance by disentangling domain features. ETL [2] captures overlapping and domain-specific attributes. II-HGCN [7] uses a hypergraph convolutional network for intra-domain and inter-domain analysis to generate accurate embeddings. Tri-CDR [19] leverages mixed behavioral sequences to capture global contexts, designing a triple cross-domain attention and contrastive learning strategy for enhanced cross-domain knowledge transfer. CrossAug [20] introduces a novel data augmentation method to effectively leverage interactions between domains.

6 Results and Analysis

6.1 RQ1: Performance Comparison

We conduct experiments on four datasets, evaluating model performance based on HR, NDCG and MRR. As shown in Table 1, DHHGCN significantly outperforms all comparative models. Notably, it achieves significant improvements over single-domain models, demonstrating that our inter-domain channel effectively aggregates shared information to enhance recommendations. Compared to various CDR models, DHHGCN also surpasses the strongest baselines in each group. The first set of experiments shows the most significant enhancements in the MRR and NDCG metrics, with increases of up to 20.63% and 18.40%, respectively. In the second set, performance on the clothing dataset shows even greater gains, with some cases exceeding 50%. These results highlight the benefits of

Table 1. Performance(%) results of two groups, in each group, the best results are marked in bold and the second-best results are underlined.

Beauty & Health												
Domain	Beauty						Health					
topK	topK = 5			topK = 10			topK = 5			topK = 10		
Metrics	HR	NDCG	MRR	HR	NDCG	MRR	HR	NDCG	MRR	HR	NDCG	MRR
Single-Domain Recommendation Methods												
LightGCN	9.68	5.46	4.31	13.74	5.21	5.83	7.96	3.75	4.71	11.18	4.65	5.35
CoHHN	18.36	12.84	11.19	25.23	15.03	11.96	18.49	13.95	12.44	23.22	15.48	13.07
BiPNet	21.56	13.66	11.06	30.11	16.46	12.23	19.30	14.09	12.38	23.70	15.54	12.99
Cross-Domain Recommendation Methods												
DisenCDR	10.43	7.04	5.92	16.06	8.87	6.69	13.06	9.12	7.82	18.84	11.09	8.73
Tri-CDR	22.18	18.77	15.93	37.42	22.02	17.28	20.54	13.73	11.50	27.48	15.97	12.42
CrossAug	22.18	15.70	13.57	31.02	16.61	14.39	23.32	16.61	14.39	30.94	19.06	15.40
PPGN	35.25	20.15	21.88	50.43	24.52	22.66	33.47	19.91	15.49	50.39	27.56	20.31
ETL	35.70	24.92	21.72	48.10	29.64	23.23	36.13	25.20	21.52	49.11	28.78	22.34
II-HGCN	36.82	25.44	21.91	50.98	29.80	23.53	36.70	25.50	21.82	48.91	29.51	23.49
DHHGCN	**41.61**	**30.12**	**26.43**	**53.12**	**33.60**	**27.83**	**40.05**	**28.45**	**23.91**	**51.32**	**32.06**	**26.62**
Improve(%)	13.00	18.40	20.63	4.20	12.75	18.27	9.13	11.57	9.58	1.85	8.64	13.32
Cloth & Sports												
Domain	Cloth						Sports					
topK	topK = 5			topK = 10			topK = 5			topK = 10		
Metrics	HR	NDCG	MRR	HR	NDCG	MRR	HR	NDCG	MRR	HR	NDCG	MRR
Single-Domain Recommendation Methods												
LightGCN	2.80	1.21	1.59	4.06	1.52	1.81	6.28	2.75	3.65	9.57	3.39	3.94
CoHHN	10.25	7.07	6.03	14.00	6.53	8.28	8.12	5.73	5.05	11.19	6.73	5.41
BiPNet	23.67	15.12	12.27	33.67	17.54	13.24	10.38	6.11	4.79	16.38	8.05	5.60
Cross-Domain Recommendation Methods												
DisenCDR	5.29	3.11	2.41	16.31	9.03	2.64	6.64	4.28	3.51	11.12	5.84	4.25
Tri-CDR	5.83	3.56	2.82	9.71	4.78	3.31	9.32	6.06	5.00	14.21	7.64	5.66
CrossAug	5.81	4.28	3.82	8.55	5.81	4.91	19.14	13.63	11.81	26.27	15.93	12.76
PPGN	10.35	3.04	4.71	10.76	5.41	6.72	10.12	4.59	5.94	18.58	6.03	8.91
ETL	24.91	18.00	18.93	35.15	21.19	20.11	29.38	21.47	19.15	39.36	24.71	20.46
II-HGCN	13.12	9.35	8.07	19.09	11.17	8.83	23.47	16.24	13.96	33.98	19.57	15.27
DHHGCN	**40.80**	**31.82**	**29.21**	**48.30**	**34.54**	**30.33**	**34.37**	**24.72**	**22.07**	**43.51**	**27.96**	**23.34**
Improve(%)	63.79	76.77	54.3	37.41	63.00	50.80	16.98	15.14	15.25	10.54	13.15	14.08

our model's intra-domain handling of heterogeneous item information and inter-domain aggregation of shared knowledge, leading to substantial improvements across all metrics.

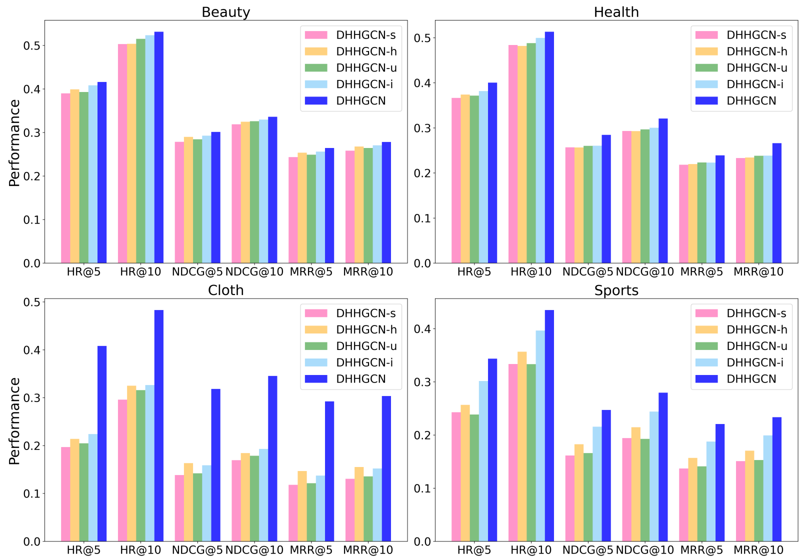

Fig. 3. The presentation of DHHGCN ablation experiments on four datasets, where DHHGCN-s retains only the intra-domain modeling part, DHHGCN-h removes the intra-domain heterogeneous attribute module, DHHGCN-u and DHHGCN-i remove the inter-domain user and item modules, respectively.

6.2 RQ2: Ablation Study

To validate the design components of DHHGCN, we conduct four ablation experiments focusing on intra-domain heterogeneous attribute information and inter-domain information aggregation, as shown in Fig. 3. DHHGCN achieves the best performance across all metrics. Specifically, DHHGCN-s, which retains only intra-domain modeling, shows significant performance fluctuations due to the lack of inter-domain information aggregation and transfer. However, it still outperforms other models in NDCG and MRR, demonstrating the strength of its intra-domain modeling. DHHGCN-h, which excludes item attribute information, performs notably worse, highlighting the importance of heterogeneous attributes in mitigating data sparsity. DHHGCN-u and DHHGCN-i, which retain only the user or item hypergraph, respectively, also underperform compared to DHHGCN, further supporting the necessity of the inter-domain component. These results collectively validate the rationality of DHHGCN's design.

6.3 RQ3: Experiment on Sparse Datasets

To evaluate DHHGCN's effectiveness in sparse data scenarios, we sparsify the dataset by reducing each user's interaction records by 10%, 30%, 50% and compare it with two top-performing CDR models, ETL and II-HGCN. As shown in Table 2, all models' performance declines as data sparsity increases, which

Table 2. Sparsity data experiment performance(%) of Beauty & Health, the best results are marked in bold and the second-best results are underlined.

Beauty & Health												
Domain	Beauty						Health					
topK	topK = 5			topK = 10			topK = 5			topK = 10		
Metrics	HR	NDCG	MRR	HR	NDCG	MRR	HR	NDCG	MRR	HR	NDCG	MRR
10%												
ETL	35.09	23.94	20.28	36.49	24.12	20.10	46.87	27.77	21.88	47.20	27.88	21.65
II-HGCN	38.47	25.36	21.07	34.93	24.19	20.86	50.25	29.23	22.66	47.40	27.97	22.46
DHHGCN	**40.78**	**29.43**	**25.90**	**37.75**	**26.68**	**23.17**	**50.52**	**32.65**	**27.05**	**49.48**	**30.59**	**24.78**
Improve(%)	6.0	16.05	22.92	3.45	10.29	11.07	0.54	11.7	19.37	4.39	9.37	10.33
30%												
ETL	34.36	23.26	19.62	32.05	22.08	18.27	46.17	27.07	21.11	42.75	25.99	19.89
II-HGCN	34.62	22.48	18.50	31.48	22.38	19.37	45.12	26.83	20.32	43.19	26.60	20.51
DHHGCN	**40.22**	**29.01**	**25.37**	**35.06**	**25.59**	**22.61**	**48.58**	**31.79**	**26.46**	**46.60**	**29.41**	**24.26**
Improve(%)	16.18	24.72	29.31	9.39	14.34	16.73	5.22	17.44	25.34	7.9	10.56	18.28
50%												
ETL	32.42	20.66	16.80	29.37	18.49	14.91	43.20	24.08	18.24	39.67	21.75	16.27
II-HGCN	31.81	21.16	17.69	28.94	19.33	16.45	44.13	25.18	19.37	42.03	23.70	18.26
DHHGCN	**39.55**	**28.39**	**24.71**	**33.13**	**23.35**	**20.36**	**50.06**	**32.11**	**26.25**	**45.93**	**27.72**	**22.17**
Improve(%)	22.0	34.17	39.68	12.22	20.8	23.77	17.58	27.52	35.52	9.28	16.96	21.41

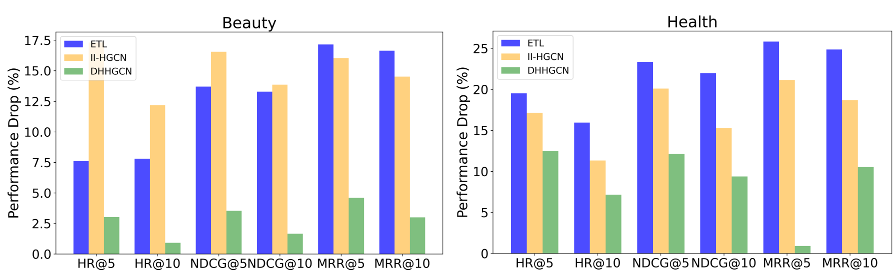

Fig. 4. The reduction ratios of each model on various evaluation metrics after data sparsity processing.

aligns with expectations since fewer interactions limit the model's ability to capture user preferences. However, DHHGCN demonstrates greater robustness, with a smaller performance drop compared to the baselines. Fig. 4 further illustrates this: after 50% sparsification, II-HGCN's metrics drop by over 10% on average, while ETL shows reductions ranging from 7.6% to 25.82%. In contrast, DHHGCN's performance remains stable, with reductions ranging from just 0.91% to 12.23%, highlighting its superior ability to handle sparse data.

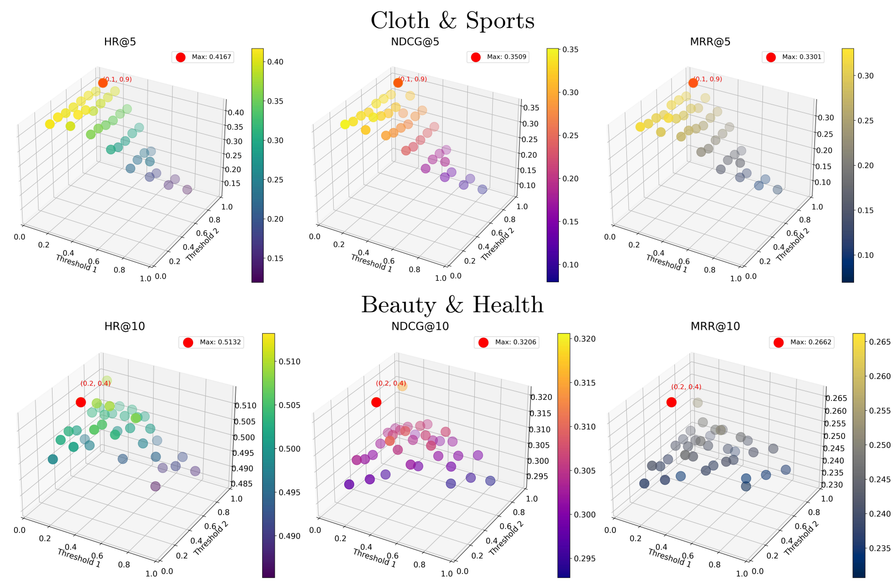

Fig. 5. Results of HR@5/10, NDCG@5/10, MRR@5/10 for Cloth & Sports and Beauty & Health under varying thresholds. The best results are highlighted in red font.

6.4 RQ4: Parameter Analysis

To construct inter-domain hypergraphs for users and items, we create a similarity ranking matrix by setting threshold values, which reduces complexity and enhances effective information propagation during convolution. We conduct experiments across a range of potential threshold values for two experimental groups, as shown in Fig. 5, all metrics exhibit nonlinear gradients. To reduce noise and improve feature capture for long-tail users and items, we set a threshold t_1 to filter weakly similar nodes, experiments reveal that setting t_1 closer to 0 often yields optimal performance, with best values typically around 0.1 and 0.2. The threshold t_2 helps in distinguishing the similarity strength among similar nodes, which is useful for aggregating node information by assigning weights based on similarity levels. We find that when $t_1 = 0.1$, the best performance occurs at $t_2 = 0.9$; for $t_1 = 0.2$, performance improves with $t_2 = 0.4$. A higher threshold prioritizes highly similar nodes, benefiting specific metrics but not overall effectiveness, whereas a lower threshold enhances differentiation and leads to uniform improvements. Based on this, we set $t_1 = 0.2$ and $t_2 = 0.4$ for optimal balance.

6.5 RQ5: Complexity Analysis

In DHHGCN, we identify the inter-type convolution on the heterogeneous hypergraph is the most time-consuming part, with a time complexity of $O((n+k)\times r \times \bar{N}_t)$, where n and k are the number of items in different domains, r is the number of iterations, $\bar{N}_t$ represents the average neighbor counts for each attributes (ID, price and category). Experiments show that DHHGCN achieves higher recommendation accuracy but takes slightly longer to compute than baseline models. Specifically, the best baseline, II-HGCN, averages 20-25 seconds per epoch, while DHHGCN takes 25-30 seconds. Although trading computational cost for accuracy is reasonable, we plan to improve efficiency in future work. For example, focusing only on attributes of items more relevant to user preferences may reduce computational overhead and enhance recommendation interpretability.

7 Conclusion and Future Work

In this paper, we propose a Dual-Channel Heterogeneous Hypergraph Convolutional Network for CDR. It constructs hypergraph structures for users and items in both intra-domain and cross-domain channels, leveraging diverse convolutional algorithms to capture their fine-grained high-order relationships. By incorporating an element-wise gating mechanism, it effectively balances domain-specific knowledge and cross-domain shared knowledge, improving recommendation performance. Experiments show DHHGCN outperforms state-of-the-art CDR methods, especially in sparse data scenarios.

In the future, we aim to extend DHHGCN to more recommendation tasks. Its core lies in intra-domain and cross-domain hypergraph convolution, which captures fine-grained high-order relationships between the subjects and targets of recommendation. This domain-agnostic design adapts to various data types and non-overlapping CDR scenarios. Additionally, we will optimize computational efficiency and integrate techniques like graph prompt learning to improve model's interpretability.

Acknowledgments. This work is supported by Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Cao, J., Lin, X., Cong, X., Ya, J., Liu, T., Wang, B.: Disencdr: Learning disentangled representations for cross-domain recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22). pp. 267–277. Association for Computing Machinery, New York, NY, USA (2022)
2. Chen, X., Zhang, Y., Tsang, I.W., Pan, Y., Su, J.: Toward equivalent transformation of user preferences in cross domain recommendation. ACM Transactions on Information Systems (TOIS) **41**(1), 1–31 (January 2023)
3. Chen, Y., Wang, Y., Ni, Y., Zeng, A.X., Lin, L.: Scenario-aware and mutual-based approach for multi-scenario recommendation in e-commerce. In: Proceedings of the 2020 International Conference on Data Mining Workshops (ICDMW). pp. 127–135 (2020)

4. Feng, Y., You, H., Zhang, Z., Ji, R., Gao, Y.: Hypergraph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 3558–3565 (2019)
5. Guo, L., Tang, L., Chen, T., Zhu, L., Nguyen, Q.V.H., Yin, H.: Da-gcn: domain-aware attentive graph convolution network for shared-account cross-domain sequential recommendation. arXiv preprint arXiv:2105.03300 (2021)
6. Guo, X., Li, S., Guo, N., Cao, J., Liu, X., Ma, Q., Gan, R., Zhao, Y.: Disentangled representations learning for multi-target cross-domain recommendation. ACM Transactions on Information Systems **41**(4), 85:1–85:27 (March 2023)
7. Han, Z., Zheng, X., Chen, C., Cheng, W., Yao, Y.: Intra and inter domain hypergraph convolutional network for cross-domain recommendation. In: Proceedings of the ACM Web Conference 2023. pp. 449–459. WWW '23, Association for Computing Machinery, New York, NY, USA (2023)
8. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: Lightgcn: simplifying and powering graph convolution network for recommendation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 639–648. Association for Computing Machinery, New York, NY, USA (2020)
9. Hu, L., Cao, J., Xu, G., Cao, L., Gu, Z., Zhu, C.: Personalized recommendation via cross-domain triadic factorization. In: Proceedings of the 22nd International Conference on World Wide Web. pp. 595–606. WWW '13, Association for Computing Machinery, New York, NY, USA (2013)
10. Ji, S., Feng, Y., Ji, R., Zhao, X., Tang, W., Gao, Y.: Dual channel hypergraph collaborative filtering. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 2020–2029. KDD '20, Association for Computing Machinery, New York, NY, USA (2020)
11. Cui, Q., Yang, F., Zhang, X., et al.: HeroGRAPH: A Heterogeneous Graph Framework for Multi-Target Cross-Domain Recommendation. In: ORSUM@RecSys 2020. LNCS, vol. 12645, pp. 45–60. Springer, Cham (2020)
12. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2017)
13. Li, P., Tuzhilin, A.: Ddtcdr: Deep dual transfer cross domain recommendation. In: Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM 2020). pp. 331–339. WSDM '20, Association for Computing Machinery, New York, NY, USA (2020)
14. Lian, J., Zhang, F., Xie, X., Sun, G.: Cccfnet: A content-boosted collaborative filtering neural network for cross domain recommender systems. In: Proceedings of the 26th International Conference on World Wide Web Companion. pp. 817–818. WWW '17 Companion, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (2017)
15. Liu, J., Zhao, P., Zhuang, F., Liu, Y., Sheng, V.S., Xu, J., Zhou, X., Xiong, H.: Exploiting aesthetic preference in deep cross networks for cross-domain recommendation. In: Proceedings of the Web Conference 2020. pp. 2768–2774. WWW '20, Association for Computing Machinery, New York, NY, USA (2020)
16. Liu, M., Li, J., Li, G., Pan, P.: Cross domain recommendation via bi-directional transfer graph collaborative filtering networks. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management. pp. 885–894 (2020)
17. Liu, Y., Zhu, S., Xia, J., Ma, Y., Ma, J., Zhong, W., Liu, X., Yu, S., Zhang, K.: End-to-end learnable clustering for intent learning in recommendation. In: Proc. of NeurIPS (2024)

18. Liu, Y., Zhu, S., Yang, T., Ma, J., Zhong, W.: Identify then recommend: Towards unsupervised group recommendation. In: Proc. of NeurIPS (2024)
19. Ma, H., Xie, R., Meng, L., Chen, X., Zhang, X., Lin, L., Zhou, J.: Triple sequence learning for cross-domain recommendation. ACM Transactions on Information Systems (ACM Trans. Inf. Syst.) **42**(4), 91:1–91:29 (February 2024)
20. Mao, Q., Liu, Q., Li, Z., Wu, L., Lv, B., Zhang, Z.: Cross-reconstructed augmentation for dual-target cross-domain recommendation. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2352–2356. SIGIR '24, ACM, New York, NY, USA (2024)
21. Sun, X., Cheng, H., Liu, B., Li, J., Chen, H., Xu, G., Yin, H.: Self-supervised hypergraph representation learning for sociological analysis. IEEE Transactions on Knowledge and Data Engineering **35**(11), 11860–11871 (2023)
22. Sun, X., Yin, H., Liu, B., Chen, H., Cao, J., Shao, Y., Viet Hung, N.Q.: Heterogeneous hypergraph embedding for graph classification. In: Proceedings of the 14th ACM international conference on web search and data mining. pp. 725–733 (2021)
23. Wang, Y., Feng, C., Guo, C., Chu, Y., Hwang, J.N.: Solving the sparsity problem in recommendations via cross-domain item embedding based on co-clustering. In: Proceedings of the 12th ACM International Conference on Web Search and Data Mining (WSDM 2019). pp. 717–725. ACM, New York, NY, USA (2019)
24. Xu, J., Wang, X., Zhang, H., Lv, P.: Heterogeneous and clustering-enhanced personalized preference transfer for cross-domain recommendation. Information Fusion **99**, 101892 (2023)
25. Xie, Y., Yu, C., Jin, X., et al.: Heterogeneous graph contrastive learning for cold start cross-domain recommendation. Knowledge-Based Systems **299**, 112054 (2024)
26. Xu, Z., Wei, P., Liu, S., Zhang, W., Wang, L., Zheng, B.: Correlative preference transfer with hierarchical hypergraph network for multi-domain recommendation. In: Proceedings of the ACM Web Conference 2023. pp. 983–991 (2023).
27. Yadati, N., Nimishakavi, M., Yadav, P., Nitin, V., Louis, A., Talukdar, P.: Hypergcn: A new method for training graph convolutional networks on hypergraphs. Advances in Neural Information Processing Systems **32**, 482–493 (2019)
28. Yu, J., Yin, H., Li, J., Wang, Q., Hung, N.Q.V., Zhang, X.: Self-supervised multi-channel hypergraph convolutional network for social recommendation. In: Proceedings of the Web Conference 2021 (WWW '21). pp. 413–424. Association for Computing Machinery, New York, NY, USA (2021)
29. Xia, X., Yin, H., Yu, J., Wang, Q., Cui, L., Zhang, X.: Self-supervised hypergraph convolutional networks for session-based recommendation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35(5), pp. 4503–4511. AAAI Press (2021)
30. Zang, T., Zhu, Y., Liu, H., Zhang, R., Yu, J.: A survey on cross-domain recommendation: Taxonomies, methods, and future directions. ACM Transactions on Information Systems **41**(2), 42:1–42:44 (2022)
31. Zhang, X., Xu, B., Ma, F., Li, C., Lin, Y., Lin, H.: Bi-preference learning heterogeneous hypergraph networks for session-based recommendation. ACM Transactions on Information Systems (ACM TOIS) **42**(3), 1–28 (2023)
32. Zhang, X., Xu, B., Yang, L., Li, C., Ma, F., Liu, H., Lin, H.: Price does matter! modeling price and interest preferences in session-based recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2022). pp. 1684–1693. ACM, New York, NY, USA (2022)

33. Zhao, C., Li, C., Fu, C.: Cross-domain recommendation via preference propagation graphnet. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. pp. 2165–2168 (2019)
34. Zheng, Y., Gao, C., He, X., Li, Y., Jin, D.: Price-aware recommendation with graph convolutional networks. In: 2020 IEEE 36th International Conference on Data Engineering (ICDE). pp. 133–144 (2020)
35. Zhou, D., Huang, J., Schölkopf, B.: Learning with hypergraphs: Clustering, classification, and embedding. Advances in Neural Information Processing Systems **19** (2006)
36. Zhu, F., Chen, C., Wang, Y., Liu, G., Zheng, X.: Dtcdr: A framework for dual-target cross-domain recommendation. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM 2019). pp. 1533–1542. Beijing, China (2019)
37. Zhu, F., Wang, Y., Chen, C., Liu, G., Zheng, X.: A graphical and attentional framework for dual-target cross-domain recommendation. In: IJCAI. vol. 21, p. 39 (2020)
38. Song, Z., Zhang, W., Deng, L., Zhang, J., Wu, Z., Bian, K., Cui, B.: Mitigating negative transfer in cross-domain recommendation via knowledge transferability enhancement. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 2745–2754. ACM (2024)
39. Zhu, F., Wang, Y., Chen, C., Zhou, J., Li, L., Liu, G.: Cross-domain recommendation: challenges, progress, and prospects. In: arXiv preprint arXiv:2103.01696. arXiv, Cornell University Library (2021)
40. Zhu, F., Wang, Y., Zhou, J., Chen, C., Li, L., Liu, G.: A unified framework for cross-domain and cross-system recommendations. IEEE Transactions on Knowledge and Data Engineering **35**(2), 1171–1184 (2023)