Social Recommendation via Graph-Level Counterfactual Augmentation

Yinxuan Huang[1*], Ke Liang[1*],
Yanyi Huang[1], Xiang Zeng[1], Kai Chen[1†], Bin Zhou[1†]
[1]College of Computer Science and Technology
National University of Defense Technology
Changsha, Hunan, China
{huangyinxuan22,chenkai_,huangyanyi22,binzhou}@nudt.edu.cn, liangke200694@126.com

Abstract

Traditional recommendation system focus more on the correlations between users and items (user-item relationships), while research on user-user relationships has received significant attention these years, which is also known as social recommendation. Graph-based models have achieved a great success in this task by utilizing the complex topological information of the social networks. However, these models still face the insufficient expressive and overfitting problems. Counterfactual approaches are proven effective as information augmentation strategies towards above issues in various scenarios, but not fully utilized in social recommendations. To this end, we propose a novel social recommendation method, termed SR-GCA, via a plug-and-play Graph-Level Counterfactual Augmentation mechanism. Specifically, we first generate counterfactual social and item links by constructing a counterfactual matrix for data aug- mentation. Then, we employ a supervised learning strategy to refine data both factual and counterfactual links. Thirdly, we enhance representations learning between users via an alignment and self-supervised optimization techniques. Extensive experiments demonstrate the promising capacity of our model from five aspects: superiority, effectiveness, transferability, complexity, and robustness. In particular, the transferability is well-proven by extending our GCA module to three typical social recommendation models.

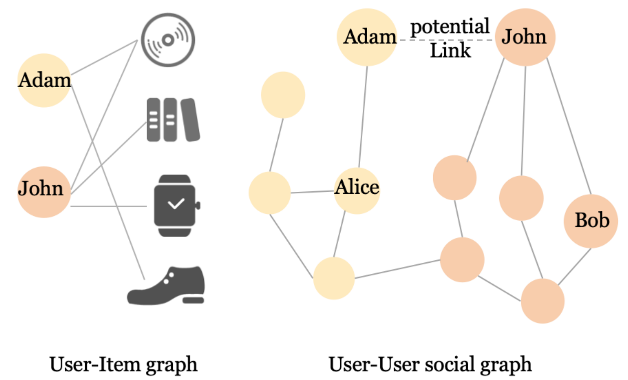

Figure 1: A comparison of user-user interaction graphs and user-item bipartite graphs. Traditional RS tasks generally focus on simpler user-item interactions within User-Item graph, whereas SR delves into the more complex relationships within User-User social Graphs.

Introduction

Recommendation systems (RS) primarily emphasize the correlations between users and items (user-item relationships), achieving remarkable success on e-commerce platforms like Amazon, Taobao, and others (Tu et al. 2021; Yu et al. 2022). However, as the amount of available information increases and more people prefer to make friends online, the scope of recommendation scenarios has expanded from user-item interactions to user-user interactions (Mei, Huang, and Li 2021) (see Fig. 1). This shift has led to the emergence of a new field known as social recommendation (SR), which focuses on exploring the more complex relationships between users, in contrast to traditional RS tasks that typically deal with simpler user-item interactions in bipartite graph structures. A multitude of SR methods have been proposed, such as SocialMF (Qian 2015) and SoRec (Ma et al. 2008a; Liang et al. 2024a), which extract features by decomposing rating and social matrices or applying social regularization. Recently, the rapid evolution of graph neural networks (GNNs) has prompted exploration of their application within the SR domain (Qin et al. 2020; Liang et al. 2024b). This interest stems from GNNs' capability to synthesize node-specific information and the underlying topological structures (Sun et al. 2024a; Liang et al. 2024c). Their adeptness at managing the complexities embedded in social relationship networks positions GNNs as an ideal choice for modeling the non-Euclidean nature of social dynamics and user interactions (Qin et al. 2021; Song et al. 2024; Luo et al. 2024a). They naturally represent user interactions and user-item relationships as graph-structured data. GNN-based SR methods typically follow two approaches: either integrating user-item and user-user graphs into a single heterogeneous graph for unified representation learning or separately modeling these graphs and combining the resulting vectors for a comprehensive user representation (Yu

[*] Co-first authors with equally contribution.
[†] Corresponding authors.

et al. 2022; Liu et al. 2023b; Sun et al. 2024b). However, graph data sparsity remains a challenge, often leading to insufficient expressiveness and overfitting(Cai et al. 2023; Ni et al. 2024; Luo et al. 2023). To this end, graph data augmentation, aiming to enrich the feature space and simulate diverse social interactions, has become a focal point in the realm of SR research (Xia et al. 2023; Zhang et al. 2024).

In recent years, the development of causal inference theory has provided strong support for improving the performance of recommendation systems, including debiasing data, data augmentation, and model enhancement, as well as how to achieve the interpretability, diversity, and fairness of recommendation systems (Zhu, Ma, and Li 2023). Notably, counterfactual inference lies at the core of causal inference, and counterfactual data augmentation has demonstrated significant advantages (Yu et al. 2024; Tu et al. 2024b,a), enabling the exploration of alternative scenarios and enhancing models' ability to discern causal effects, it enables the exploration of alternative scenarios and strengthens the models' ability to understand causality and make more informed recommendations. However, current counterfactual-enhanced SR models primarily focus on user-item interactions through negative sampling (Bayer, Kaufhold, and Reuter 2022; Ren et al. 2024), neglecting richer user-user and item-item associations, and thus overlooking significant graph structural information (Zhao et al. 2022; Liang et al. 2023). For example, in Fig.1, *Alice* and *Adam* share common interests, likely due to community ties. Similar interests can also be found between individuals from different communities, like *Adam* and *John* may share common interests, represented by a potential link. Although these potential links may seem intuitively reasonable and informative, they might not be recorded for various reasons, such as privacy concerns or limitations in time, resources, and budget. From the perspective of causal inference, these unrecorded potential links are actually counterfactual links (Abrate and Bonchi 2021; Bajaj et al. 2021; Tan et al. 2022). As an important supplementary resource to observed graph, these counterfactual links can be more comprehensively utilized at the graph level, thereby helping to train SR models with better accuracy and generalization capabilities. To our best knowledge, this research direction has yet to be thoroughly explored.

To address this gap, we propose a novel method called SR-GCA, which generates counterfactual graph-level links to enhance SR. By posing counterfactual questions such as "*Would Alice and Adam be socially connected without their shared community?*" in Fig. 1, we can understand the underlying causality and fully capture the complex topology of the graph. Therefore, we propose applying counterfactual learning at the graph level in SR to fully leverage the graph-level structural information. These counterfactual links represent unobservable outcomes for a given pair of nodes under hypothetical conditions. By creating counterfactual links for all positive and negative training examples, we effectively perform graph data augmentation, enriching the training set. Specifically, we first generate counterfactual graph-level links from the observable graph. Then, we train a GNN-based link predictor to learn user representations that predict both factual and counterfactual interactions, enabling us to evaluate individual treatment effects. Ultimately, the predictor aims to identify the key factors driving interactions within the graph structure. Our contributions are as follows:

1. We provide a new perspective on the causal relationship in graph-level structure by exploring counterfactual questions. SR-GCA facilitates counterfactual data augmentation in SR, highlighting the advantages of graph-level counterfactual augmentation over negative sampling.
2. We introduce SR-GCA, a novel module leverages the causal relationship between social network structure and social links to enhance SR. While item-item and user-user graphs are complex and rich in information, the user-item graph requires less detail at the graph level.
3. Extensive experiments highlight the strong capabilities of our model across five key aspects: superiority, effectiveness, transferability, complexity, and Robustness. Notably, the model's transferability is demonstrated by integrating GCA module into three different SR models.

Related Work

Social-Aware Recommendation

Numerous social recommendation methodologies (Huang et al. 2019; Liu et al. 2023a) integrate online social relationships among users into the recommendation framework as side information. Early methods (Ma et al. 2008b; Guo, Zhang, and Yorke-Smith 2016) rely on matrix factorization to project users into latent factors, based on the assumption that users often share interests with their socially affiliated counterparts. With the rise of deep learning, many recent works (Shen et al. 2020; Luo et al. 2024b) have applied deep learning techniques to social recommendation tasks, achieving promising results. More recently, studies (Wu et al. 2019; Huang et al. 2021) have explored the application of GCNs to simultaneously model both user-user and user-item relationships. Moreover, attention mechanisms have been introduced to discern variations in user influence, thereby better capturing their preferences (Fan et al. 2019). Additionally, some methods (Long et al. 2021; Yu et al. 2021) incorporate graph augmentation into contemporary SR to further enhance performance. However, previous approaches have rarely fully exploited the rich information within the complex topological structures of graphs structures.

Counterfactual Learning on Graphs

The counterfactual paradigm, central to causal inference, estimates causal effects by exploring "What if" scenarios. Counterfactual learning has gained widespread adoption in various machine learning fields recently, including computer vision (Abbasnejad et al. 2020) and natural language processing (Calderon et al. 2022). This approach helps reduce overfitting, offering more robust and generalized models (Kaddour et al. 2022). To avoid false explanations and identify key causal factors in predictions, researchers (Bajaj et al. 2021; Tan et al. 2022) have developed various models to obtain counterfactual explanations on graphs. Counterfactual link prediction is widely studied in the contexts of link prediction tasks (Zhao et al. 2022), knowledge graph completion (Zhao et al. 2022), and RS (Mu et al. 2022; Song et al.

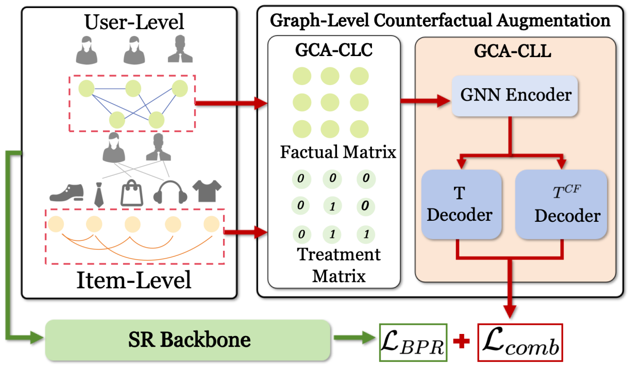

Figure 2: An overview of SR-GCA. We input user-item graph in Social Recommendatioin (SR) Backbone and User-level graph into Graph-Level Counterfactual Augmentation Module.

2023) to explore the root causes of link formation. However, few studies have explored counterfactual learning from a graph-level perspective in SR.

Methodology

Preliminary

Given a user behavior dataset in a social recommendation scenario, which includes interactions between N users $\mathcal{U} = \{u_1, ..., u_N\}$ and M items $\mathcal{V} = \{v_1, ..., v_M\}$, we represent the user-item interaction data using the collaborative graph $\mathcal{G}_r = (\mathcal{U}, \mathcal{V}, \mathcal{E}_r)$, where edges are formed when a user u_i interacts with an item v_j. The user-item interaction matrix $\mathcal{E}_r = [x_{i,j}] \in \mathbb{R}^{N\times M}$ is defined such that $x_{i,j} \in \{0, 1\}$, with $x_{i,j} = 1$ if the interaction (u_i, v_j) is observed and $x_{i,j} = 0$ otherwise. To encode user-user social relations, we define the user-user social graph $\mathcal{G}_s = (\mathcal{U}, \mathcal{E}_s)$, where $\mathcal{U}$ represents the set of users. The adjacency matrix $\mathcal{E}_s \in \{0, 1\}^{N\times N}$ indicates social connections between users, with $\mathcal{E}_s(i, j) = 1$ denoting a connection between user u_i and user u_j. Additionally, to represent potential counterfactual links between items, we define the item-item interaction graph $\mathcal{G}_i = (\mathcal{V}, \mathcal{E}_i)$, where $\mathcal{V} = \{v_1, v_2, \ldots, v_M\}$ is the set of items. The adjacency matrix $\mathcal{E}_i \in \{0, 1\}^{M\times M}$ encodes these relationships, with $\mathcal{E}_i(i, j) = 1$ indicating a counterfactual connection between item v_i and item v_j.

Overview

The proposed method consists of two main components: the SR Lightweight-GCN Backbone and the Graph-Level Counterfactual Augmentation (SR-GCA). SR-GCA enhances RS by generating counterfactual links to improve link prediction. Specifically, it comprises two key modules: the Counterfactual Links Constructor (GCA-CLC) and the Counterfactual Links Learner (GCA-CLL). By employing lightweight graph convolutional networks (GCNs) and a simple MLP decoder, this method refines both factual and counterfactual relationships, thereby enhancing expressiveness and reducing overfitting. We input a User-User graph to obtain user predictions.

Notation	Explanation
$U = \{u_1, \ldots, u_N\}$	Set of users
$V = \{v_1, \ldots, v_M\}$	Set of items
$\mathcal{G}_r = (U, V, \mathcal{E}_r)$	The user-item collaborative graph
$\mathcal{G}_s = (U, \mathcal{E}_s)$	The user-user social graph
$\mathcal{G}_i = (V, \mathcal{E}_i)$	The item-item interaction graph
A	Factual link matrix
A^{CF}	Counterfactual link matrix
$\mathbf{T}$	Factual treatment matrix
$\mathbf{T}^{CF}$	Counterfactual treatment matrix
$T_{i,j}$	Binary treatment entry for (u_i, u_j)

Table 1: Notation Summary

Social Recommendation (SR) Backbone

Inspired by the effectiveness of lightweight GCN-enhanced collaborative filtering paradigms, we adopt it as our backbone model to configure the user-item interaction graphs:

$$\mathcal{E}_r^{(l)} = (\mathcal{L}_r + I) \cdot \mathcal{E}_r^{(l-1)}, \tag{1}$$

where $\mathcal{E}_r^{(l)}, \mathcal{E}_r^{(l-1)} \in \mathbb{R}^{(I+J)\times d}$ are the embeddings of users and items after l iterations of modeling. $I \in \mathbb{R}^{(I+J)\times(I+J)}$ denotes the identity matrix to enable self-loops. The Laplacian matrix $\mathcal{L}_r \in \mathbb{R}^{(I+J)\times(I+J)}$ is defined as:

$$\mathcal{L}_r = \mathbf{D}_r^{-\frac{1}{2}} \mathbf{A}_r \mathbf{D}_r^{-\frac{1}{2}}, \quad \mathbf{A}_r = \begin{bmatrix} 0 & \mathcal{E} \\ \mathcal{E}^\top & 0 \end{bmatrix}, \tag{2}$$

here, $\mathcal{E} \in \mathbb{R}^{I\times J}$ is the user-item interaction matrix, and 0 represents an all-zero matrix. The bidirectional adjacency matrix $\mathbf{A}_r$ of the user-item interaction view is normalized by multiplying it with the corresponding diagonal degree matrix $\mathbf{D}_r$. Additionally, to capture user-user social relations, we apply a lightweight GCN to the user social graph $\mathcal{G}_s$. Each layer of this lightweight GCN is defined as follows:

$$\mathcal{E}_s^{(l)} = (\mathcal{L}_s + I) \cdot \mathcal{E}_s^{(l-1)}, \tag{3}$$

$$\mathcal{L}_s = \mathbf{D}_s^{-\frac{1}{2}} \mathbf{A}_s \mathbf{D}_s^{-\frac{1}{2}}, \tag{4}$$

here, $\mathbf{A}_s \in \mathbb{R}^{I\times I}$ encodes the social connections between users, and $\mathbf{D}_s, \mathcal{L}_s \in \mathbb{R}^{I\times I}$ are the corresponding diagonal degree matrix and the normalized Laplacian matrix, respectively. The embeddings $\mathbb{E}_s^{(l)}, \mathbb{E}_s^{(l-1)} \in \mathbb{R}^{I\times d}$ represent the user embeddings at the l-th and $(l-1)$-th layers of the graph neural network, respectively.

To aggregate the embeddings encoded from different orders in $\mathcal{G}_r$ and $\mathcal{G}_s$, our model adopts mean-pooling operators for both the interaction and social views, L is the maximum number of graph iterations.:

$$\bar{\mathcal{E}}_r = \sum_{l=0}^{L} \mathcal{E}_r^{(l)}, \quad \bar{\mathcal{E}}_s = \sum_{l=0}^{L} \mathcal{E}_s^{(l)}. \tag{5}$$

For item-item links, we first construct an item-item graph to capture the semantic similarities between items. The embedding of item i is initialized from its representation in the user-item collaboration graph, which is the embedding obtained in the context of interactions between the item and users. For clarity, we denote this initial embedding as e_0^i:

$$e_0^i = e_0^u \odot \sigma(e_0^u \times W^i + b^i), \tag{6}$$

where e_0^u is obtained after the embedding propagation on the social graph, W^i and b^i are learnable parameters representing the weight matrix and bias vector, respectively, σ is an activation function (commonly the sigmoid function), and $\odot$ denotes element-wise multiplication. We adopt the same embedding propagation method as in the user-user graph, using LightGCN for processing.

Graph-Level Counterfactual Augmentation (GCA)

The Graph-Level Counterfactual Augmentation comprises two modules: GCA-CLC and GCA-CLL. The user-user graph and item-item graph are constructed in the same manner. In this context, we primarily use the user-user graph as an example to demonstrate the process.

Counterfactual Links Constructor (GCA-CLC) We denote the user-user adjacency matrix as $\mathcal{E}_s$. For simplicity, we represent the factual outcomes as A, and the unobserved matrix of counterfactual links as A^{CF}, which represents the counterfactual outcomes when the treatment differs. We denote $T \in \{0,1\}^{N\times N}$ as the binary factual treatment matrix, where $T_{i,j}$ indicates the treatment of user pairs (u_i, u_j). We define T^{CF} as the counterfactual treatment matrix, where each element $T_{i,j}^{CF} = 1 - T_{i,j}$. Our objective is to estimate the counterfactual outcomes A and A^{CF} (representing the observed and counterfactual data) to enhance link prediction.

Since we cannot observe the potential outcomes under the opposite treatment, we aim to find the nearest neighbor observed context as a substitute. First, we compute the similarity of node pairs using node-level embeddings, as calculating the distance between all pairs of nodes is extremely inefficient and impractical in application. We learn the node embedding $\tilde{X} \in \mathbb{R}^{N\times D}$. Therefore, for each $(v_i, v_j) \in \mathcal{V} \times \mathcal{V}$, we define its counterfactual link (v_a, v_b) as:

$$(v_a, v_b) = \arg\min_{v_a, v_b \in \mathcal{V}} \{d(\tilde{x}_i, \tilde{x}_a) + d(\tilde{x}_j, \tilde{x}_b)\} \tag{7}$$

$$\text{s.t.} \quad T_{a,b} = 1 - T_{i,j}, \quad d(\tilde{x}_i, \tilde{x}_a) + d(\tilde{x}_j, \tilde{x}_b) < 2\gamma, \tag{8}$$

where $d(\cdot,\cdot)$ is specified as the Euclidean distance in the embedding space of $\tilde{X}$, and γ is the hyperparameter that defines the maximum distance at which two nodes are considered similar. If a node pair is not found within this distance, we do not assign any nearest neighbor for the given node pair to ensure all neighbors are sufficiently similar in the feature space. Thus, if $\exists(v_a, v_b) \in \mathcal{V} \times \mathcal{V}$, the treatment matrix and counterfactual adjacency matrix are defined as:

$$T_{i,j}^{CF} = 1 - T_{i,j}, \quad A_{i,j}^{CF} = A_{a,b}, \tag{9}$$

otherwise, they are defined as:

$$T_{i,j}^{CF} = T_{i,j}, \quad A_{i,j}^{CF} = A_{i,j}. \tag{10}$$

Algorithm 1: GCA Algorithm

Input: $f, g, \mathbf{A}, \mathbf{X}, n\_epochs, n\_epoch\_ft$
Output: $\hat{\mathbf{A}}$ for factual link matrix, $\hat{\mathbf{A}}^{CF}$ for counterfactual link matrix

```
Compute the treatment matrix T using Eqs. (7) and (8).
Compute the counterfactual treatment matrix T^CF and the counterfactual link matrix A^CF using Eqs. (9) and (10).
for epoch ∈ range(n_epochs) do
    Compute the embedding matrix Z = f(A, X).
    Generate Â and Â^CF using the function g and Eqs. (11) and (12).
    Update the parameters Θ_f and Θ_g by minimizing the loss function L (Eq. 16).
end for
Freeze Θ_f and re-initialize Θ_g.
Compute the embedding matrix Z = f(A, X).
for epoch ∈ range(n_epoch_ft) do
    Generate Â using the function g and Eqs. (11) and (12).
    Update Θ_g by minimizing the loss function L_F (Eq. 16).
end for
Compute the embedding matrix Z = f(A, X).
Generate Â and Â^CF using the function g and Eqs. (11) and (12).
```

Counterfactual Links Learner (GCA-CLL) We represent each node pair using the Hadamard product of their respective vectors. Specifically, for a node pair (v_i, v_j), the representation is $z_i \circ z_j \in \mathbb{R}^H$, where $\circ$ denotes the Hadamard product. Drawing inspiration from (Zhao et al. 2022), we adopt a straightforward approach by employing a simple decoder based on a multi-layer perceptron (MLP). This decoder leverages the representations of node pairs and their treatments. The function g is defined as follows:

$$\hat{A} = g(Z, T), \text{s.t.} \hat{A}_{i,j} = \text{MLP}([z_i \circ z_j, T_{i,j}]), \tag{11}$$

$$\hat{A}^{CF} = g(Z, T^{CF}), \text{s.t.} \hat{A}_{i,j}^{CF} = \text{MLP}([z_i \circ z_j, T_{i,j}^{CF}]), \tag{12}$$

here, $[\cdot,\cdot]$ denotes vector concatenation, and the outputs $\hat{A}$ and $\hat{A}^{CF}$ are used to estimate the observed Individual Treatment Effect (ITE) and the Average Treatment Effect (ATE). For each node pair (v_i, v_j), the ITE is calculated. The ITE and ATE are calculated as follows:

$$\text{ATE} = \mathbb{E}_{z\sim Z}\text{ITE}(z), \tag{13}$$

$$\text{ITE}(z) = g(z, 1) - g(z, 0), \tag{14}$$

$$\text{ITE}_{(v_i, v_j)} = g((z_i, z_j), 1) - g((z_i, z_j), 0), \tag{15}$$

and we use those ITE and ATE improve the learning of Z. The loss function are as follows:

$$\begin{aligned}
\mathcal{L}_F &= \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} A_{i,j} \cdot \log \hat{A}_{i,j} \\
&\quad + (1 - A_{i,j}) \cdot \log(1 - \hat{A}_{i,j}), \\
\mathcal{L}_{CF} &= \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} A_{i,j}^{CF} \cdot \log \hat{A}_{i,j}^{CF} \\
&\quad + (1 - A_{i,j}^{CF}) \cdot \log(1 - \hat{A}_{i,j}^{CF}),
\end{aligned} \tag{16}$$

To align the factual and counterfactual distribution representations, we introduce discrepancy distance as an additional loss term to regularize representation learning. This loss minimizes the distance between the learned representations from $\hat{P}^F$ and $\hat{P}^{CF}$,

$$\mathcal{L}_{\text{dist}} = \|\hat{P}_f^F - \hat{P}_f^{CF}\|_F \,, \tag{17}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. The combined learning objective is defined as:

$$\mathcal{L}_{comb} = \mathcal{L}_F + \alpha \cdot \mathcal{L}_{CF} + \beta \cdot \mathcal{L}_{dist}. \tag{18}$$

Here, α and β are hyperparameters that control the weights of the counterfactual outcome estimation loss and the discrepancy loss, respectively. The final user embedding is obtained from the last layer of the MLP, with the same process applied to items.

Loss Function

SR-GCA employs the Bayesian Personalized Ranking (BPR) loss (Wang, Xia, and Huang 2023) to optimize the counterfactual graph-augmented social relationships, thereby enhancing recommendation quality. BPR maximizes the user's preference for positive samples, enabling the model to learn a more personalized ranking structure. The loss function is defined as follows:

$$\mathcal{L}_{BPR} = \sum_{(u,v^+,v^-)} -\log\left(\sigma(\hat{r}_{u,v^+} - \hat{r}_{u,v^-})\right) \tag{19}$$

where (u, v^+, v^-) is a triplet sample for pairwise recommendation training, with v^+ as a positive item user u has interacted with and v^- as a randomly sampled negative item from non-interacted ones. The function $\sigma(.)$ denotes the Sigmoid activation, while $\hat{r}_{u,v^+}$ and $\hat{r}_{u,v^-}$ are the predicted scores for the positive and negative items, respectively. The predicted score $\hat{r}_{u,v}$ is derived from embeddings obtained via counterfactual graph augmentation, given by:

$$\hat{r}_{u,v} = \tilde{\mathbf{e}}_u^\top \mathbf{e}_v^r \tag{20}$$

To further improve model robustness and generalization, SR-GCA integrates BPR loss with counterfactual inference loss in a joint optimization framework. The final objective function is given by:

$$\mathcal{L} = \mathcal{L}_{BPR} + \lambda \mathcal{L}_{comb} \tag{21}$$

where λ is a hyperparameter used to balance the trade-off between ranking optimization and counterfactual reasoning enhancement, effectively leverages counterfactual augmentation to enable the model to better capture user preferences, thereby improving the recommendation system's accuracy and generalization capabilities.

Disccusion

GCA between User-User and Item-Item rather than User-Item. The need to capture complex topological relationships has been a significant focus in previous research. While it is essential to understand and supplement relationships within complex topological structures, the similarity between users and items often lacks meaningful relevance. However, this similarity becomes critically important when dealing with similar objects, where it plays a crucial role in enhancing the accuracy and effectiveness of the model. Therefore, it is imperative to explore methods that not only consider the topological complexities but also effectively leverage the similarities in objects of the same kind to improve overall performance.

Data	Ciao	Epinions
# Users	6,672	11,111
# Items	98,875	190,774
# Interactions	198,181	247,591
Interaction Density	0.0300%	0.0117%
# Social Ties	109,503	203,989
Social Ties Density	0.246%	0.165%

Table 2: Statistics of the two datasets.

Extension on our SR-GCA with Different Backbones. The SR-GCA module offers strong scalability as a plug-and-play component that can be seamlessly integrated into other SR backbones. We performed experiments to validate its transferability, with detailed results available in the Experiments section (RQ2) .

Experiments

In this section, we present a series of experiments conducted to evaluate the performance of our SR-GCA model, focusing on the following research questions (RQs):

- **RQ1:** How does SR-GCA compare to other state-of-the-art methods?
- **RQ2:** How effective is the transferability of the GCA?
- **RQ3:** What impact do counterfactual links and various treatments have on our method's performance?
- **RQ4:** Is SR-GCA sufficiently robust to handle noisy and sparse data in SR?
- **RQ5:** How does the time complexity of our method compared to alternative approaches?

Experimental Settings

Datasets and Evaluation Metrics We conducted experiments using two widely-used real-world datasets: **Ciao** and **Epinions**. Detailed statistics for these datasets are presented in Tab. 2. For evaluation, we employ Hit Ratio HR@N and Normalized Discounted Cumulative Gain (NDCG)@N as metrics, where N represents the number of items recommended to the user, with a default value of 10 (Wang, Xia, and Huang 2023).

Baselines We conducted comparative analyses against 10 state-of-the-art recommendation models: PMF (Salakhutdinov and Mnih 2007), TrustMF (Yang et al. 2013), DiffNet (Wu et al. 2019), DGRec (Song et al. 2019), EATNN (Chen et al. 2019), NGCF (Wang et al. 2019), MHCN (Yu et al. 2021), KCGN (Huang et al. 2021), SMIN (Long et al. 2021), DSL (Wang, Xia, and Huang 2023).

Method	Ciao		Epinions	
	HR@N	NDCG	HR@N	NDCG
PMF [2007]	0.4223	0.2464	0.1686	0.0968
TrustMF [2017]	0.4492	0.2520	0.1769	0.0842
DiffNet [2019]	0.5544	0.3167	0.2182	0.2055
DGRec [2019]	0.4658	0.2401	0.2055	0.0908
EATNN [2019]	0.4255	0.2525	0.1576	0.0794
NGCF [2019]	0.5629	0.3429	0.2969	0.1582
MHCN [2021]	0.5950	0.3805	0.3507	0.1926
KCGN [2021]	0.5785	0.3552	0.3122	0.1721
SMIN [2021]	0.5852	0.3687	0.3159	0.1867
DSL [2023]	0.6374	0.4065	0.3983	0.2290
SR-GCA	**0.6586**	**0.4262**	**0.4120**	**0.2455**

Table 3: Comprehensive analysis of overall performance.

Implementation Details The SR-GCA model was implemented in PyTorch and optimized using the Adam optimizer. The learning rate was tuned within $[5e^{-4}, 1e^{-3}, 5e^{-3}]$ with a 0.96 decay factor per epoch. Batch sizes were selected from [1024, 2048, 4096, 8192], and hidden dimensions from [64, 128, 256, 512]. The Graph-Level Counterfactual Augmentation module employed a GNN encoder and a 3-layer MLP decoder with a 64-dimensional hidden layer and ELU activation. The parameter γ was set according to the γ_{pct}-percentile of node embedding distances for each dataset. The optimal number of GNN layers was chosen from [1, 2, 3, 4]. Regularization weights λ was selected from $[1e^{-3}, 1e^{-2}, 1e^{-1}, 1e^{0}, 1e^{1}]$.

Overall Performance (RQ1)

An exhaustive comparative analysis of SR-GCA's performance relative to existing methods (see Tab. 3) demonstrates that SR-GCA consistently outperforms all baseline methods.Specifically, on the Ciao dataset, SR-GCA achieves HR@N of 0.6586, surpassing the closest competitor, DSL, by an impressive margin of 8.72%. Additionally, the NDCG score reaches 0.4262, marking a significant 7.38% improvement over the best-performing baseline, MHCN. On the Epinions dataset, SR-GCA also excels, achieving an HR@N of 0.4120, which represents a substantial 5.93% improvement over the next best method MHCN, the NDCG score of 0.2455 for our model, reflecting an 6.22% improvement over DSL, underscores the distinct advantages of our counterfactual augmentation strategy. More specifically, SR-GCA model performed better on the Ciao dataset, primarily due to its higher density of social ties compared to the Epinions dataset (See Tab. 2). The increased density leads to richer user-item interactions, allowing the model to learn more effectively and make more accurate predictions. This denser structure enables the model to capture underlying patterns in the data more effectively, resulting in superior performance on the Ciao dataset.

Model	Ciao		Epinions	
	HR@N	**NDCG**	**HR@N**	**NDCG**
GDMSR	0.5402	0.3201	0.7219	0.5022
+ GCA	0.5543	0.3449	0.7392	0.5183
DGNN	0.5515	0.3383	0.7335	0.5215
+ GCA	0.5703	0.3528	0.7403	0.5309

Table 4: Performances of GCA on DGNN and GDMSR

Method	Ciao		Epinions	
	HR@N	NDCG	HR@N	NDCG
SR-GCA	0.6586	0.4262	0.4220	0.2455
SR-GCA w/o CL of items	0.6203	0.4023	0.3993	0.2308
SR-GCA w/o CL of users	0.6329	0.4102	0.4010	0.2388
SR-GCA w/o CL of both	0.6091	0.3872	0.3375	0.2012

Table 5: Ablation Study of GCA.

Transferability Analysis (RQ2)

We validated the effectiveness of the GCA module by extracting it from the SR-GCA model and then integrating it into two baseline models, DGNN and GDMSR, for further evaluation. The experimental results (see Tab. 4) indicate that GCA enhances the performance of both DGNN and GDMSR, making it a valuable addition to these SR-backbone models. Specifically, the GCA module improved DGNN's HR@N by approximately 3% on the Ciao dataset and by 2% on the Epinions dataset, while it improved GDMSR's HR@N by around 2% on Ciao and by 2% on Epinions. The GCA model performed better when integrated with DGNN compared to GDMSR, primarily because DGNN also employs a contrastive learning structure, which complements the GCA module's strengths. This synergy enhances the model's ability to learn from counterfactual examples, leading to superior performance. All percentages are rounded to the nearest whole number.

Ablation Study (RQ3)

Impact of GCA on Different Graph Level. We conducted an ablation study to assess the impact of counterfactual links on our GCA model (see Tab. 5). The variations include **SR-GCA w/o CL of items** (no counterfactual links between items), **SR-GCA w/o CL of users** (no counterfactual links between users), and **SR-GCA w/o CL of both** (no counterfactual links between users and items). The significant performance improvements observed underscore the critical role of counterfactual links in enhancing the SR-GCA model, emphasizing the importance of considering hypothetical social connections alongside actual data.

Impact of Different Treatment Strategies. Moreover, we compared different treatments on the Ciao dataset, with a specific focus on various graph clustering methods (see Fig. 3). Among all those methods, the k-core clustering method demonstrates the best performance with an HR@N of 0.6586. This superior performance could be attributed to k-core's ability to more effectively identify and preserve the most influential and densely connected subgraphs within the network, which enhances the model's capacity to learn from the most relevant interactions, leading to more accurate predictions compared to other clustering methods.

Robustness Evaluation (RQ4)

Data Noise. To test the model's resilience to noise, we introduced synthetic edges at varying levels (10%, 20%, 30%) to mimic a noisy social graph (see Fig. 4). Our method demonstrates robustness against competitors. Specifically, SR-GCA maintained a relative HR of 0.95 at a 10% noise ratio on the Ciao dataset, which is approximately 3% higher than MHCN and 9% higher than NGCF+. On the Epinions dataset, SR-GCA retained a relative HR@N of 0.90 at the same noise level, outperforming MHCN by about 5% and NGCF+ by 8%. This advantage stems from two factors: the generation of enriched and diverse social data through counterfactual augmentation and a causal model that discerns causality over correlation.

Data Sparsity. To assess performance for less active users, we divided users into four groups based on interaction levels: 0–5, 6–10, 11–20, and $\geq$21 interactions, and compared recommendation accuracy across these groups. As shown in Fig. 4, SR-GCA consistently outperforms MHCN and NGCF+, particularly in low-interaction scenarios common in real-world settings. For instance, in the Ciao dataset, SR-GCA maintains a high HR@10 of approximately 0.64 for users with 0–5 interactions, outperforming MHCN by 8% and NGCF+ by 12%. In the Epinions dataset, SR-GCA achieves an HR@N of about 0.41 for the same group, surpassing MHCN by 10% and NGCF+ by 14%. SR-GCA remains robust across all sparsity levels, while other models exhibit more variability.

Complexity Analysis (RQ5)

Experimental results (Fig. 5(b)) show SR-GCA outperforms most methods in inference efficiency. On the Ciao dataset, SR-GCA completes each epoch in 5 seconds, 59.6% faster than average and 37.5% faster than SMIN (8 seconds). On Epinions, it runs in 7 seconds per epoch, 55.6% faster than average and better than DSL. This efficiency improvement primarily affects the inference phase. By using counterfactual augmentation, SR-GCA generates more informative social links during training, enabling better aggregation of neighbor information during inference, reducing redundant computations, and accelerating convergence.

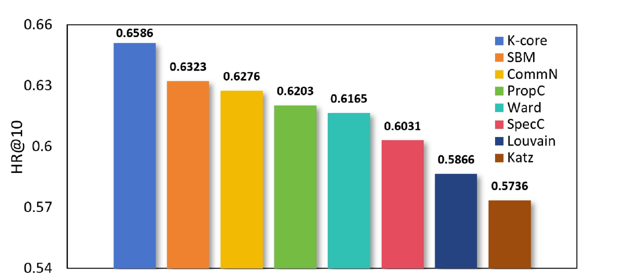

Figure 3: Results of GCA with different treatments.

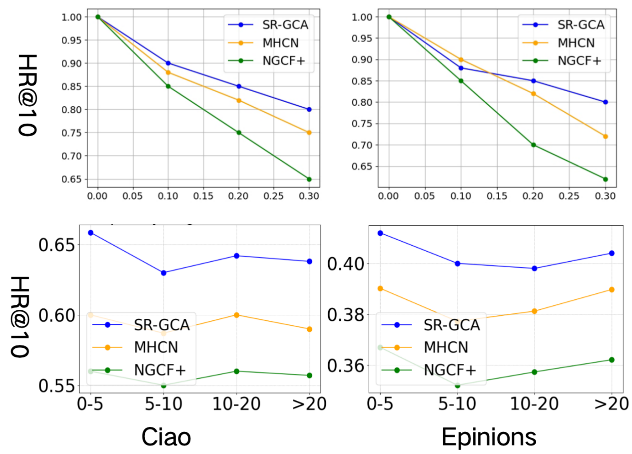

Figure 4: Robustness study with respect to data noise and sparsity: the upper plots represent data noise, while the lower ones depict data sparsity.

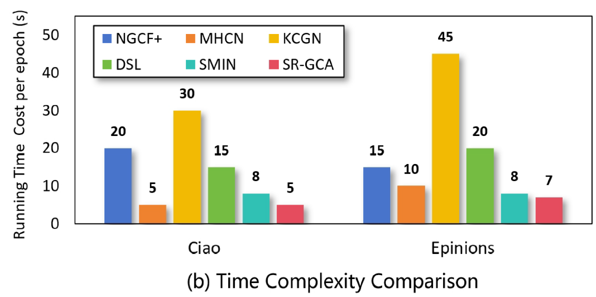

Figure 5: Complexity Analysis.

Conclusion

In this paper, we propose **SR-GCA**, a novel social recommendation method that leverages counterfactual link generation to enhance link prediction and uncover causal relationships within social graphs. By introducing counterfactual social links, SR-GCA effectively facilitates data augmentation, improving the robustness and accuracy of social recommendations. While counterfactual approaches have demonstrated their effectiveness across various domains, their potential in social recommendation remains largely unexplored. Extensive experiments validate the superiority of SR-GCA, highlighting its ability to enhance recommendation performance through counterfactual reasoning.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments and helpful suggestions. This work was supported by the National Natural Science Foundation of China (No.62172428)

References

Abbasnejad, E.; Teney, D.; Parvaneh, A.; Shi, J.; and van den Hengel, A. 2020. Counterfactual Vision and Language Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 10041–10051. IEEE.

Abrate, C.; and Bonchi, F. 2021. Counterfactual Graphs for Explainable Classification of Brain Networks. In Zhu, F.; Ooi, B. C.; and Miao, C., eds., *KDD 2021*, 2495–2504. ACM.

Bajaj, M.; Chu, L.; Xue, Z. Y.; Pei, J.; Wang, L.; Lam, P. C.; and Zhang, Y. 2021. Robust Counterfactual Explanations on Graph Neural Networks. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y. N.; Liang, P.; and Vaughan, J. W., eds., *Proc. of NeurIPS*, 5644–5655.

Bayer, M.; Kaufhold, M.-A.; and Reuter, C. 2022. A Survey on Data Augmentation for Text Classification. *ACM Computing Surveys*, 55(7): 1–39.

Cai, D.; Qian, S.; Fang, Q.; Hu, J.; Ding, W.; and Xu, C. 2023. Heterogeneous Graph Contrastive Learning Network for Personalized Micro-Video Recommendation. *IEEE Trans. Multim.*, 25: 2761–2773.

Calderon, N.; Ben-David, E.; Feder, A.; and Reichart, R. 2022. DoCoGen: Domain Counterfactual Generation for Low Resource Domain Adaptation. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proc. of ACL*, 7727–7746. Dublin, Ireland: ACL.

Chen, C.; Zhang, M.; Wang, C.; Ma, W.; Li, M.; Liu, Y.; and Ma, S. 2019. An Efficient Adaptive Transfer Neural Network for Social-aware Recommendation. In Piwowarski, B.; Chevalier, M.; Gaussier, É.; Maarek, Y.; Nie, J.; and Scholer, F., eds., *Proc. of SIGIR*, 225–234. ACM.

Fan, W.; Ma, Y.; Li, Q.; He, Y.; Zhao, Y. E.; Tang, J.; and Yin, D. 2019. Graph Neural Networks for Social Recommendation. In Liu, L.; White, R. W.; Mantrach, A.; Silvestri, F.; McAuley, J. J.; Baeza-Yates, R.; and Zia, L., eds., *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, 417–426. ACM.

Guo, G.; Zhang, J.; and Yorke-Smith, N. 2016. A Novel Recommendation Model Regularized with User Trust and Item Ratings. *TKDE*, 28(7): 1607–1620.

Huang, C.; Xu, H.; Xu, Y.; Dai, P.; Xia, L.; Lu, M.; Bo, L.; Xing, H.; Lai, X.; and Ye, Y. 2021. Knowledge-aware Coupled Graph Neural Network for Social Recommendation. In *AAAI*, 4115–4122. AAAI Press.

Huang, S.; Kang, Z.; Tsang, I. W.; and Xu, Z. 2019. Auto-weighted multi-view clustering via kernelized graph learning. *Pattern Recognition: The Journal of the Pattern Recognition Society*, 88.

Kaddour, J.; Lynch, A.; Liu, Q.; Kusner, M. J.; and Silva, R. 2022. Causal Machine Learning: A Survey and Open Problems. *ArXiv preprint*, abs/2206.15475.

Liang, K.; Liu, Y.; Zhou, S.; Tu, W.; Wen, Y.; Yang, X.; Dong, X.; and Liu, X. 2023. Knowledge graph contrastive learning based on relation-symmetrical structure. *IEEE Transactions on Knowledge and Data Engineering*, 36(1): 226–238.

Liang, K.; Liu, Y.; Zhou, S.; Tu, W.; Wen, Y.; Yang, X.; Dong, X.; and Liu, X. 2024a. Knowledge Graph Contrastive Learning Based on Relation-Symmetrical Structure. *IEEE Transactions on Knowledge and Data Engineering*, 36(1): 226–238.

Liang, K.; Meng, L.; Liu, M.; Liu, Y.; Tu, W.; Wang, S.; Zhou, S.; Liu, X.; Sun, F.; and He, K. 2024b. A survey of knowledge graph reasoning on graph types: Static, dynamic, and multi-modal. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Liang, K.; Meng, L.; Zhou, S.; Tu, W.; Wang, S.; Liu, Y.; Liu, M.; Zhao, L.; Dong, X.; and Liu, X. 2024c. MINES: Message Intercommunication for Inductive Relation Reasoning over Neighbor-Enhanced Subgraphs. In *AAAI Conference on Artificial Intelligence*.

Liu, Z.; Chen, Z.; Zhang, M.; Duan, S.; Wen, H.; Li, L.; Li, N.; Gu, Y.; and Yu, G. 2023a. Modeling User Viewing Flow Using Large Language Models for Article Recommendation.

Liu, Z.; Mei, S.; Xiong, C.; Li, X.; Yu, S.; Liu, Z.; Gu, Y.; and Yu, G. 2023b. Text Matching Improves Sequential Recommendation by Reducing Popularity Biases.

Long, X.; Huang, C.; Xu, Y.; Xu, H.; Dai, P.; Xia, L.; and Bo, L. 2021. Social Recommendation with Self-Supervised Metagraph Informax Network. In Demartini, G.; Zuccon, G.; Culpepper, J. S.; Huang, Z.; and Tong, H., eds., *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, 1160–1169. ACM.

Luo, Y.; Li, H.; Liu, Q.; Shi, L.; and Wu, X.-M. 2024a. Node Identifiers: Compact, Discrete Representations for Efficient Graph Learning.

Luo, Y.; Li, H.; Shi, L.; and Wu, X.-M. 2023. Enhancing Graph Transformers with Hierarchical Distance Structural Encoding.

Luo, Y.; Liu, Q.; Shi, L.; and Wu, X.-M. 2024b. Structure-aware Semantic Node Identifiers for Learning on Graphs. *CoRR*, abs/2405.16435.

Ma, H.; Yang, H.; Lyu, M. R.; and King, I. 2008a. Sorec: social recommendation using probabilistic matrix factorization. *ACM*.

Ma, H.; Yang, H.; Lyu, M. R.; and King, I. 2008b. SoRec: social recommendation using probabilistic matrix factorization. In Shanahan, J. G.; Amer-Yahia, S.; Manolescu, I.; Zhang, Y.; Evans, D. A.; Kolcz, A.; Choi, K.; and Chowdhury, A., eds., *Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, Napa Valley, California, USA, October 26-30, 2008*, 931–940. ACM.

Mei, D.; Huang, N.; and Li, X. 2021. Light Graph Convolutional Collaborative Filtering With Multi-Aspect Information. *IEEE Access*, 9: 34433–34441.

Mu, S.; Li, Y.; Zhao, W. X.; Wang, J.; Ding, B.; and Wen, J. 2022. Alleviating Spurious Correlations in Knowledge-aware Recommendations through Counterfactual Generator. In Amigó, E.; Castells, P.; Gonzalo, J.; Carterette, B.;

Culpepper, J. S.; and Kazai, G., eds., *Proc. of SIGIR*, 1401–1411. ACM.

Ni, X.; Xiong, F.; Zheng, Y.; and Wang, L. 2024. Graph Contrastive Learning with Kernel Dependence Maximization for Social Recommendation. In Chua, T.; Ngo, C.; Kumar, R.; Lauw, H. W.; and Lee, R. K., eds., *WWW 2024*, 481–492. ACM.

Qian, Y. 2015. Matrix factorization recommender algorithm using trust propagation. *Journal of Chongqing University of Arts and Sciences*.

Qin, L.; Li, Z.; Che, W.; Ni, M.; and Liu, T. 2021. Co-GAT: A Co-Interactive Graph Attention Network for Joint Dialog Act Recognition and Sentiment Classification. In *AAAI*, 13709–13717. AAAI Press.

Qin, L.; Xu, X.; Che, W.; and Liu, T. 2020. AGIF: An Adaptive Graph-Interactive Framework for Joint Multiple Intent Detection and Slot Filling. In Cohn, T.; He, Y.; and Liu, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2020*, 1807–1816. Online: Association for Computational Linguistics.

Ren, X.; Xia, L.; Yang, Y.; Wei, W.; Wang, T.; Cai, X.; and Huang, C. 2024. SSLRec: A Self-Supervised Learning Framework for Recommendation. In Caudillo-Mata, L. A.; Lattanzi, S.; Medina, A. M.; Akoglu, L.; Gionis, A.; and Vassilvitskii, S., eds., *WSDM 2024*, 567–575. ACM.

Salakhutdinov, R.; and Mnih, A. 2007. Probabilistic Matrix Factorization. In Platt, J. C.; Koller, D.; Singer, Y.; and Roweis, S. T., eds., *NeurIPS*, 1257–1264. Curran Associates, Inc.

Shen, T.; Jia, J.; Li, Y.; Ma, Y.; Bu, Y.; Wang, H.; Chen, B.; Chua, T.; and Hall, W. 2020. PEIA: Personality and Emotion Integrated Attentive Model for Music Recommendation on Social Media Platforms. In *AAAI*, 206–213. AAAI Press.

Song, W.; Wang, S.; Wang, Y.; Liu, K.; Liu, X.; and Yin, M. 2023. A Counterfactual Collaborative Session-based Recommender System. In Ding, Y.; Tang, J.; Sequeda, J. F.; Aroyo, L.; Castillo, C.; and Houben, G., eds., *WWW 2023*, 971–982. ACM.

Song, W.; Xiao, Z.; Wang, Y.; Charlin, L.; Zhang, M.; and Tang, J. 2019. Session-Based Social Recommendation via Dynamic Graph Attention Networks. In Culpepper, J. S.; Moffat, A.; Bennett, P. N.; and Lerman, K., eds., *WSDM 2019*, 555–563. ACM.

Song, Y.; Gu, Y.; Li, T.; Qi, J.; Liu, Z.; Jensen, C. S.; and Yu, G. 2024. CHGNN: A Semi-Supervised Contrastive Hypergraph Learning Network. *TKDE*, 36(9): 4515–4530.

Sun, Y.; Qin, Y.; Li, Y.; Peng, D.; Peng, X.; and Hu, P. 2024a. Robust Multi-View Clustering With Noisy Correspondence. *IEEE Transactions on Knowledge and Data Engineering*, 36(12): 9150–9162.

Sun, Y.; Ren, Z.; Hu, P.; Peng, D.; and Wang, X. 2024b. Hierarchical Consensus Hashing for Cross-Modal Retrieval. *IEEE Transactions on Multimedia*, 26: 824–836.

Tan, J.; Geng, S.; Fu, Z.; Ge, Y.; Xu, S.; Li, Y.; and Zhang, Y. 2022. Learning and Evaluating Graph Neural Network Explanations based on Counterfactual and Factual Reasoning. In Laforest, F.; Troncy, R.; Simperl, E.; Agarwal, D.; Gionis, A.; Herman, I.; and Médini, L., eds., *WWW 2022*, 1018–1027. ACM.

Tu, W.; Liao, Q.; Zhou, S.; Peng, X.; Ma, C.; Liu, Z.; Liu, X.; Cai, Z.; and He, K. 2024a. RARE: Robust Masked Graph Autoencoder. *TKDE*, 36(10): 5340–5353.

Tu, W.; Zhou, S.; Liu, X.; Ge, C.; Cai, Z.; and Liu, Y. 2024b. Hierarchically Contrastive Hard Sample Mining for Graph Self-supervised Pre-training. *TNNLS*, 35(11): 16748–16761.

Tu, W.; Zhou, S.; Liu, X.; Guo, X.; Cai, Z.; Zhu, E.; and Cheng, J. 2021. Deep Fusion Clustering Network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9978–9987.

Wang, T.; Xia, L.; and Huang, C. 2023. Denoised Self-Augmented Learning for Social Recommendation. In *IJCAI 2023*, 2324–2331. ijcai.org.

Wang, X.; He, X.; Wang, M.; Feng, F.; and Chua, T. 2019. Neural Graph Collaborative Filtering. In Piwowarski, B.; Chevalier, M.; Gaussier, É.; Maarek, Y.; Nie, J.; and Scholer, F., eds., *Proc. of SIGIR*, 165–174. ACM.

Wu, L.; Sun, P.; Fu, Y.; Hong, R.; Wang, X.; and Wang, M. 2019. A Neural Influence Diffusion Model for Social Recommendation. In Piwowarski, B.; Chevalier, M.; Gaussier, É.; Maarek, Y.; Nie, J.; and Scholer, F., eds., *Proc. of SIGIR*, 235–244. ACM.

Xia, L.; Shao, Y.; Huang, C.; Xu, Y.; Xu, H.; and Pei, J. 2023. Disentangled Graph Social Recommendation.

Yang, B.; Lei, Y.; Liu, D.; and Liu, J. 2013. Social Collaborative Filtering by Trust. In Rossi, F., ed., *IJCAI 2013*, 2747–2753. IJCAI/AAAI.

Yu, J.; Yin, H.; Li, J.; Wang, Q.; Hung, N. Q. V.; and Zhang, X. 2021. Self-Supervised Multi-Channel Hypergraph Convolutional Network for Social Recommendation. In Leskovec, J.; Grobelnik, M.; Najork, M.; Tang, J.; and Zia, L., eds., *Proc. of WWW*, 413–424. ACM / IW3C2.

Yu, Z.; Feng, B.; He, D.; Wang, Z.; Huang, Y.; and Feng, Z. 2024. LG-GNN: Local-Global Adaptive Graph Neural Network for Modeling Both Homophily and Heterophily. In *International Joint Conference on Artificial Intelligence*.

Yu, Z.; Jin, D.; Wei, J.; Liu, Z.; Shang, Y.; Xiao, Y.; Han, J.; and Wu, L. 2022. TeKo: Text-Rich Graph Neural Networks with External Knowledge.

Zhang, Q.; Xia, L.; Cai, X.; Yiu, S.; Huang, C.; and Jensen, C. S. 2024. Graph Augmentation for Recommendation.

Zhao, T.; Liu, G.; Wang, D.; Yu, W.; and Jiang, M. 2022. Learning from Counterfactual Links for Link Prediction. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvári, C.; Niu, G.; and Sabato, S., eds., *Proc. of ICML*, volume 162 of *Proceedings of Machine Learning Research*, 26911–26926. PMLR.

Zhu, Y.; Ma, J.; and Li, J. 2023. Causal Inference in Recommender Systems: A Survey of Strategies for Bias Mitigation, Explanation, and Generalization. arXiv:2301.00910.