Multi-Graph Heterogeneous Interaction Fusion for Social Recommendation

CHENGYUAN ZHANG, College of Computer Science and Electronic Engineering, Hunan University
YANG WANG, School of Computer Science and Information Engineering, Hefei University of Technology; Intelligent Interconnected Systems Laboratory of Anhui Province, Hefei University of Technology, China
LEI ZHU and JIAYU SONG, School of Computer Science and Engineering, Central South University
HONGZHI YIN, School of Information Technology and Electrical Engineering, University of Queensland

With the rapid development of online social recommendation system, substantial methods have been proposed. Unlike traditional recommendation system, social recommendation performs by integrating social relationship features, where there are two major challenges, i.e., early summarization and data sparsity. Thus far, they have not been solved effectively. In this article, we propose a novel social recommendation approach, namely Multi-Graph Heterogeneous Interaction Fusion (MG-HIF), to solve these two problems. Our basic idea is to fuse heterogeneous interaction features from multi-graphs, i.e., user–item bipartite graph and social relation network, to improve the vertex representation learning. A meta-path cross-fusion model is proposed to fuse multi-hop heterogeneous interaction features via discrete cross-correlations. Based on that, a social relation GAN is developed to explore latent friendships of each user. We further fuse representations from two graphs by a novel multi-graph information fusion strategy with attention mechanism. To the best of our knowledge, this is the first work to combine meta-path with social relation representation. To evaluate the performance of MG-HIF, we compare MG-HIF with seven states of the art over four benchmark datasets. The experimental results show that MG-HIF achieves better performance.

CCS Concepts: • **Information systems** → **Data mining**

Additional Key Words and Phrases: Social recommendation, multi-graph, meta-path, graph gan, heterogeneous interaction fusion

ACM Reference format:
Chengyuan Zhang, Yang Wang, Lei Zhu, Jiayu Song, and Hongzhi Yin. 2021. Multi-Graph Heterogeneous Interaction Fusion for Social Recommendation. *ACM Trans. Inf. Syst.* 40, 2, Article 28 (September 2021), 26 pages.
https://doi.org/10.1145/3466641

This work was supported in part by the National Natural Science Foundation of China (U1936217, 62172136, 61806035, 62072166, 61702560), the Key Research and Technology Development Projects of Anhui Province (no. 202004a5020043), and project 2018JJ3691 of Science and Technology Plan of Hunan Province.
Authors' addresses: C. Zhang, College of Computer Science and Electronic Engineering, Hunan University, Changsha, Hunan, 410082, China; email: cyzhangcse@hnu.edu.cn; Y. Wang (corresponding author), School of Computer Science and Information Engineering, Hefei University of Technology and Intelligent Interconnected Systems Laboratory of Anhui Province, Hefei University of Technology, China, Hefei, Anhui, 230009, China; email: yangwang@hfut.edu.cn; L. Zhu and J. Song, School of Computer Science and Engineering, Central South University, Changsha, Hunan, 410083, China; emails: {leizhu, jiayusong}@csu.edu.cn; H. Yin, School of Information Technology and Electrical Engineering, University of Queensland, Brisbane, Queensland, QLD 4072, Australia; email: h.yin1@uq.edu.au.

1046-8188/2021/09-ART28 $15.00
https://doi.org/10.1145/3466641

1 INTRODUCTION

In recent years, recommendation systems [5, 22] have attracted great attention [4, 68]. With the explosive growth of data in the Internet, it is particularly important to call for effective and efficient recommendation methods to filter out valuable information from massive data. Many recommendation systems integrate social networking services with recommendation services to enrich the user–item interaction data with a new social dimension. Beyond the above, users' preferences and decisions are always influenced by their friends [13]. Therefore, the performance of recommendation techniques can be significantly improved, thanks to social relation mining and representation methods [38, 52].

Graph is an effective model to represent recommendation information [6, 35, 57, 70], which can be utilized to represent users and items as vertices and their interactions, i.e., rating, purchase, comment, and so on, can be modeled as edges or relations between these vertices. Several graph-based recommendation research works have been proposed [20, 27, 42, 56, 63, 64, 69]. One group of the existing work is based on matrix factorization [32], in which the high-dimensional user–item rating matrix is decomposed into two low-dimensional user factor matrix and item factor matrix. **Matrix Factorization (MF)** aims to learn a latent presentation of users and items, whose dot product explains observed feedbacks. The other group of approaches [14, 49, 63] are the **Graph Neural Network (GNN)** based methods, which are supported by powerful feature representation of deep neural network to generate meaningful vertices representations by aggregating the interaction features from neighbors via neural networks. Furthermore, the features of the vertices can be propagated through the paths in the graphs.

Motivation. Although the existing arts can effectively improve the performance of social recommendation, there are still two major problems to be solved. The first problem is the early summarization issue [28], which means that only two vertices and the connection between them are activated, but the complex network topology and heterogeneous interaction pattern cannot be learned effectively. The second problem is data sparsity and noise [24]. Especially in large-scale recommendation or social networks, each user only interacts with a limited number of items or friends. Many similar users or items cannot be linked directly. That indicates that the latent useful relations cannot be learned effectively. A number of methods [28, 45] have been proposed to employ heterogeneous information network to model the interactive local relations to address early summarization problem. With the help of meta-path, Jin et al. [28] developed an end-to-end neighborhood-based interaction model to capture diverse semantic information. However, they ignored the significant influence of social relationships between users on recommendation. To address this limitation, we propose to extract interaction features from user–item graph and social relation network and then fuse these heterogeneous interaction information to improve the vertex representation learning, which can effectively increase the recommendation precision.

Our Method. We propose a novel social recommendation approach, called **Multi-Graph Heterogeneous Interaction Fusion (MG-HIF)**, as what is shown in Figure 1. The basic idea of MG-HIF is to fuse heterogeneous interaction information from multiple graphs, namely user–item bipartite graph and user–user social relation network to improve the vertex representation learning. To effectively solve early summarization problem and capture heterogeneous interaction information from multi-hop neighbors in user–item bipartite graph, we propose to utilize meta-path to generate several vertex sequences with different structures guided by different meta-path and then employ discrete cross-correlation to learn representations of user–item pairs. The discrete cross-correlation operation between vertex sequences can capture the influence of the interaction between multi-hop neighbors and the objective user and item, which can enhance the interaction pattern learning effectively. To capture more effective connectivity features in social relation

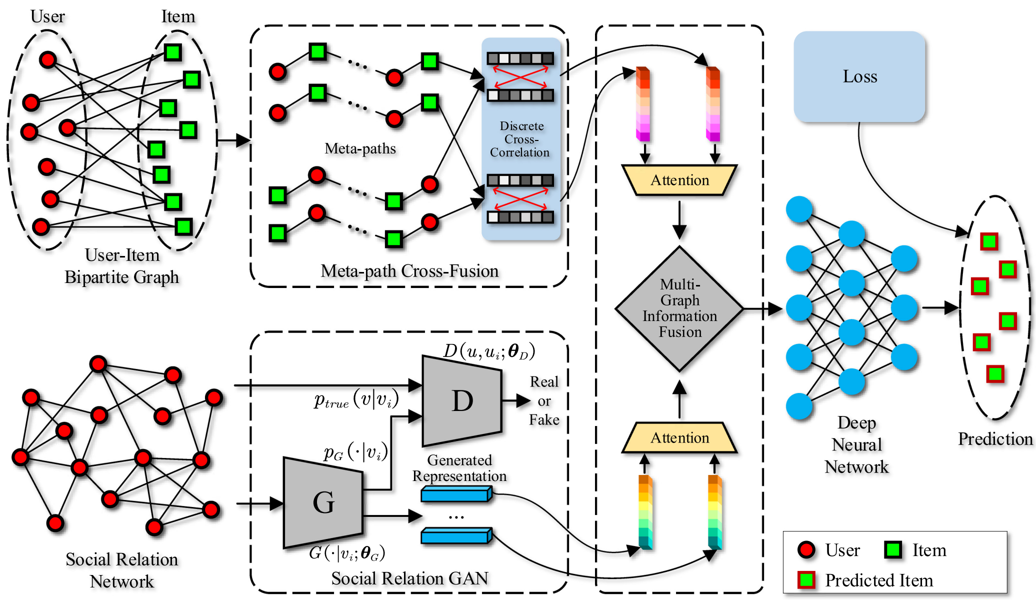


Fig. 1. The overview of the proposed MG-HIF method. It consists of three main models: (1) *A meta-path cross-fusion model* learns multi-hop relations from user–item bipartite graph; (2) *a social relation GAN model* learns the latent friendships for each user in a social network to diminish data sparsity and noise; and (3) *an attention mechanism-based multi-graph information fusion model* fuses the outputs of two above models to enhance vertex representation learning. At last, a deep neural network generates the prediction from the fused vertex representations. Best view in color.

network, we proposed an approach relaying **generative adversarial networks (GANs)** using social information or social-GAN to learn latent friendships of each user vertex, which can diminish the data sparsity and noise problem. To improve the vertex representation learning, we design a novel multi-graph information fusion model to fuse the representation of user–item pair and user vertex representation together. MG-HIF consists of two attention models, i.e., user–item attention network and user–user attention network to find the important heterogeneous interaction features from different graphs.

Contributions. The major contributions of this article are summarized as follows:

- We propose a novel approach, named MG-HIF, which fuses heterogeneous neighborhood information from meta-path cross-fusion model and social relation GAN to enhance the vertex representation capability. To the best of our knowledge, we are the first to integrate meta-path-based representation learning with adversarial learning for social recommendation.
- To fuse multi-hop heterogeneous interaction in user–item bipartite graph, we propose a meta-path cross-fusion model to capture multi-hop neighbors interaction features via discrete cross-correlations between vertex sequences produced by meta-path. Besides, to reduce the negative effects caused by data sparsity, we develop an adversarial leaning-based model, named social relation GAN, to learn latent friendships in social relation network.
- To implement heterogeneous interaction information fusion, we propose a novel multi-graph information fusion model, in which two attention techniques, i.e., user–item attention and user–user attention are used to improve the performance of feature learning.
- Comprehensive experiments are conducted over four benchmark datasets: Ciao, Filmtrust, Epinions, and Douban to evaluate MG-HIF and other baselines. The results demonstrate the superiority of MG-HIF over the states of the art.

Table 1. The Summary of Notations

Notation	Definition
$\varnothing$	1 in 1,000
$\mathcal{G}_B$	a user–item bipartite graph
$\mathcal{G}_S$	a user–user social relation network
$\mathcal{U}$	a set of user vertices
$\mathcal{V}$	a set of item vertices
$\mathcal{E}_B$	a set of edges between user and item
$\mathcal{E}_S$	a set of social relations between user pairs
u_i	the ith user
v_j	the jth item
$\mathcal{N}_U(u_i)$	the set of users that is connected to user u_i in $\mathcal{G}_S$
$\mathcal{N}_V(u_i)$	the set of items that is connected to user u_i in $\mathcal{G}_B$
$\mathcal{N}_U(v_i)$	the set of users that is connected to item v_i in $\mathcal{G}_B$
$\boldsymbol{x}_i$	the vertex embedding of user u_i
$\boldsymbol{y}_j$	the vertex embedding of item v_j
$\mathcal{A}$	the set of vertex types
$\mathcal{R}$	the set of relation types
$\mathscr{P}$	a meta-path
$\bowtie$	the concatenation operator between two objects
$\boldsymbol{\theta}$	network parameter vector
μ_i	the ith vertex in the random walk
$\mathcal{N}_{\mathscr{P}}(\mu_i)$	the set of one-hop neighbors of vertex μ_i along meta-path $\mathscr{P}$
$\mathcal{S}$	the vertex sequence set
$\otimes$	the cross-correlation operator
$\boldsymbol{M}$	the embedding matrices of vertex sequence
$\mathbb{C}$	the cross-fused representation
Λ	the number of vertex sequences
$\phi(\cdot)$	the vertex embedding function
$*$	Cartesian product operator
ξ_{ij}	the representation of user–item pairs (u_i, v_j) with attention under all meta-paths
ζ_i	the representation of user u_i with attention
z_{ij}	the fused representation
τ	the rating between user and item

Roadmap. The rest of this article is organized as follows: The problem of social recommendation and the related notions are defined in Section 2. Then we introduce our method, MG-HIF, in Section 3. Experimental results are discussed in Section 4. The related works are reviewed in Section 5. This article is finally concluded in Section 6.

2 PRELIMINARY

In this section, we introduce the definition of social recommendation and related notions in Section 2.1 and then briefly review the principle of meta-path and graph generative adversarial network in Sections 2.2 and 2.3, respectively, which are related to our method. Table 1 summarizes the frequently used mathematical notations throughout this article.

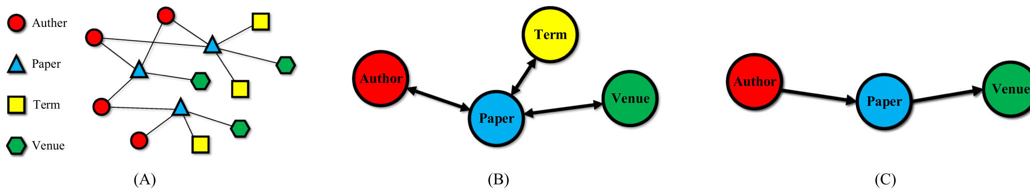


Fig. 2. An example of an information network (a), a network schema (b), and a meta-path (c).

2.1 Problem Definition

In this article, matrices and vectors are represented by bold uppercase letters and bold lowercase letters, respectively. For example, we denote a matrix with m rows and n columns as $\boldsymbol{A} \in \mathbb{R}^{m \times n}$. A ε-dimensional vector is denoted as $\boldsymbol{a} \in \mathbb{R}^{\varepsilon}$. Besides, the constants are represented by small Greek letters, such as α.

It is assumed that a social recommendation is a recommendation with online social relations (e.g., trust relations, friendships, memberships or following relations) as an additional input [30, 52]. It exploits correlations between users under the social relations to ameliorate the performance of recommendation. Let $\mathcal{G}_B = \langle \mathcal{U}, \mathcal{V}, \mathcal{E}_B \rangle$ be a user–item bipartite graph, where $\mathcal{U} = \{u_i\}_{i=1}^{m}$, $\mathcal{V} = \{v_j\}_{j=1}^{n}$ are the sets of user vertices and item vertices, respectively. Let $\mathcal{E}_B = \{e_B(u_i, v_j)\}$ be the set of edges between user vertices and item vertices and m and n represent the number of user and item vertices, respectively. Let $\mathcal{G}_S = \langle \mathcal{U}, \mathcal{E}_S \rangle$ be a user–user social relationship graph, where $\mathcal{E}_S$ denotes the relations between users. Let $\boldsymbol{B} \in \mathbb{R}^{m \times n}$ be the interaction matrix of $\mathcal{G}_B$. For each element b_{ij} in $\boldsymbol{B}$, if user u_i interacts with item v_j by rating score π, then $b_{ij} = \pi$; otherwise, $b_{ij} = 0$. Let $\boldsymbol{S} \in \mathbb{R}^{m \times m}$ be the social relationship matrix of $\mathcal{G}_S$ and each element $s_{i,k} = 1$ if the users u_i and u_k have social relation. Let $\mathcal{N}_U(u_i)$ be the set of users that connect to user u_i directly in $\mathcal{G}_S$, $\mathcal{N}_V(u_i)$ be the set of items that connect to user u_i directly in $\mathcal{G}_B$, and $\mathcal{N}_U(v_i)$ be the set of users that connected to item v_i directly in $\mathcal{G}_B$. Based on the above notation definition, we formulate the social recommendation as follows.

Definition 1 (Social Recommendation). Given two graphs $\mathcal{G}_B$ and $\mathcal{G}_S$, their corresponding relation matrices are $\boldsymbol{B}$ and $\boldsymbol{S}$. A social recommendation aims to learn a non-linear model $\mathcal{M} = \{\Phi_U, \Phi_V\}$ that generates user and item representations and predicts the mean rating between all items and a given user u_i:

$$\tau' = Predict(\Phi_U(u_i; \boldsymbol{\theta}_U), \Phi_V(v_j; \boldsymbol{\theta}_V)), u_i \in \mathcal{U}, v_j \in \hat{\mathcal{V}}, \tag{1}$$

where τ' is the mean predicted ratings of all user–item pairs. $Predict(\cdot)$ is a prediction function that is used to predict the rating between users and items. The non-linear mappings $\Phi_U(\cdot; \boldsymbol{\theta}_U)$ and $\Phi_V(\cdot; \boldsymbol{\theta}_V)$ denote the two components of social recommendation model, which is to map users and items into corresponding representation subspace, namely $\Phi_U(u_i; \boldsymbol{\theta}_U) \in \mathbb{R}^{\lambda}$ and $\Phi_V(v_j; \boldsymbol{\theta}_V) \in \mathbb{R}^{\lambda}$. $\boldsymbol{\theta}_U$ and $\boldsymbol{\theta}_V$ represent the parameter vectors of the model. To simplify the symbolic representation, we denote user u_i and item v_j representation as $\boldsymbol{x}_i$ and $\boldsymbol{y}_j$.

2.2 Meta-path

A meta-path is a specific type of path in a heterogeneous network that contains a sequence of relations defined between different types of objects (or vertices). It can be used to measure the similarities of objects in heterogeneous network. The definition of a meta-path is formulated on the notions of information network and network schema, which are reviewed, respectively, as follows. Figure 2 shows a toy example of an information network (Figure 2(a)), a network schema (Figure 2(b)), and a meta-path (Figure 2(c)), which can illustrate the topology of them.

Definition 2 (Information Network). An information network is a directed graph $\vec{\mathcal{G}} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V}$ and $\mathcal{E}$ are the set of vertices and edges. Let $\Psi : \mathcal{V} \longrightarrow \mathcal{A}$ and $\Omega : \mathcal{E} \longrightarrow \mathcal{R}$ be vertex type mapping and edge type mapping, respectively, where $\mathcal{A} = \{A_i\}_{i=1}^{p}$ and $\mathcal{R} = \{R_j\}_{j=1}^{q}$ are the set of vertex types and relation types, respectively. For each vertex $v_i \in \mathcal{V}$, there is a specific type $\Phi(v_i) \in \mathcal{A}$ corresponding to it; for each edge $e_j \in \mathcal{E}$, there is a specific type $\Omega(e_j) \in \mathcal{R}$ corresponding to it.

Definition 3 (Network Schema). A network schema is denoted as $\mathcal{T}_{\vec{\mathcal{G}}} = \langle \mathcal{A}, \mathcal{R} \rangle$, which is a meta-template of information network $\vec{\mathcal{G}} = \langle \mathcal{V}, \mathcal{E} \rangle$. Like information network, it has a vertex type mapping $\Psi : \mathcal{V} \longrightarrow \mathcal{A}$ and an edge type mapping $\Omega : \mathcal{E} \longrightarrow \mathcal{R}$.

Definition 4 (Meta-path). A meta-path $\mathscr{P}$ is a type of path defined on a network schema $\mathcal{T}_{\vec{\mathcal{G}}} = \langle \mathcal{A}, \mathcal{R} \rangle$, which is formulated as

$$\mathscr{P} := A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_{l-1}} A_l \xrightarrow{R_l} A_{l+1}, \tag{2}$$

which means a composite relation denoted as $\ddot{R} = R_1 \circ R_2 \circ \cdots \circ R_{l-1} \circ R_l$, where operator $\circ$ denotes the combination between relations.

According to Definition 4, if a path from vertex v_1 to v_{l+1} belongs to meta-path $\mathscr{P}$, then it must satisfy $\forall i, \Phi(v_i) = A_i$, and each edge $e_i(v_i, v_{i+1})$ belongs to R_i. Besides, two meta-paths $\mathscr{P}_1 = (A_1, A_2, \ldots, A_k)$ and $\mathscr{P}_2 = (A'_1, A'_2, \ldots, A'_l)$ can be concatenated together if and only if $A_k = A'_1$. The concatenated meta-path is denoted as $\mathscr{P}_{1\bowtie 2} = (A_1, A_2, \ldots, A_k, A'_2, \ldots, A'_l)$, where $\bowtie$ denotes the concatenation operator.

2.3 Graph Generative Adversarial Network

Graph generative adversarial network (GraphGAN) [54] is an extension of GAN for graph structure, which is used to generate (select) representations of latent neighbors in an adversarial manner. Similarly to GAN, a GraphGAN model consists of two components: a generator $G(\cdot; \boldsymbol{\theta}_G)$ and a discriminator $D(\cdot; \boldsymbol{\theta}_D)$. The former is to generate fake vertex representations under a true connectivity distribution, and the latter is to identify the synthetic representations from real representations via their connectivity pattern. They play a minimax game during the training and gradually enhance both of their performances.

Specifically, given a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V}$ and $\mathcal{E}$ denote the set of vertices and edges, respectively. For a vertex v_i in graph $\mathcal{G}$. Its connectivity pattern distribution is denoted as $p_{true}(v|v_i)$. Obviously, the neighbors of v_i, namely $\mathcal{N}(v_i)$, obey this distribution. The generator $G(v|v_i; \boldsymbol{\theta}_G)$ tries its best to fit the real connectivity pattern distribution $p_{true}(v_i)$ and chooses the latent neighbors that the most possibly connect with v_i to fool the discriminator. The discriminator $D(v, v_i; \boldsymbol{\theta}_D)$ receives the outputs of $G(v|v_i; \boldsymbol{\theta}_G)$ as well as v to discriminate the connectivity pattern between them is real or not. The adversarial process can be formulated as the following objective function:

$$\begin{aligned} \arg\min_{\boldsymbol{\theta}_G}\max_{\boldsymbol{\theta}_D} \mathcal{L}(\boldsymbol{\theta}_G, \boldsymbol{\theta}_D) = \sum_{i=1}^{|\mathcal{V}|} \Big(& \mathbb{E}_{v \sim p_{true}(\cdot|v_i)} \left[log D(v, v_i; \boldsymbol{\theta}_D) \right] \\ & + \mathbb{E}_{v \sim p_G(\cdot|v_i)} \left[log(1 - D(v, v_i; \boldsymbol{\theta}_D)) \right] \Big), \end{aligned} \tag{3}$$

where $|\mathcal{V}|$ denotes the number of vertices in $\mathcal{V}$ and $p_G(v|v_i)$ denotes the generated distribution of generator $G(\cdot; \boldsymbol{\theta}_G)$. The optimization of the Equation (3) can be implemented by gradient-based method as follows:

$$\theta_D^{t-1} = \theta_D^t - \eta\nabla_{\theta_D}\mathcal{L}(\theta_G, \theta_D),$$
$$\nabla_{\theta_D}\mathcal{L}(\theta_G, \theta_D) = \begin{cases} \nabla_{\theta_D} logD(v, v_i) & , if \quad v \sim p_{true} \\ \nabla_{\theta_D}(1 - logD(v, v_i)) & , if \quad v \sim p_G \end{cases} \tag{4}$$

$$\theta_G^{t-1} = \theta_G^t - \eta\nabla_{\theta_G}\mathcal{L}(\theta_G, \theta_D),$$
$$\begin{aligned} \nabla_{\theta_G}\mathcal{L}(\theta_G, \theta_D) &= \nabla_{\theta_G}\sum_{i=1}^{|\mathcal{V}|}\mathbb{E}_{v\sim p_G(\cdot|v_i)}\left[log(1 - D(v, v_i))\right] \\ &= \sum_{i=1}^{|\mathcal{V}|}\sum_{j=1}^{|\mathcal{N}(v_i)|} G(v_j|v_i)\nabla_{\theta_G} logG(v_j|v_i)log(1 - D(v_j, v_i)) \\ &= \sum_{i=1}^{|\mathcal{V}|}\mathbb{E}_{v\sim p_G(\cdot|v_i)}\left[\nabla_{\theta_G} logG(v|v_i)log(1 - D(v, v_i))\right]. \end{aligned} \tag{5}$$

3 METHODOLOGY

In this section, we propose a novel graph representation learning method for social recommendation, named the MG-HIF method, which fuses heterogeneous neighborhood information from the meta-path cross-fusion model and social relation network-based GAN to improve the vertex representation capability. We first provide an overview of MG-HIF in Section 3.1 and then discuss meta-path cross-fusion, social relation GAN, and multi-graph information fusion in Sections 3.2, 3.3, and 3.4, respectively. Finally, the optimization of this model is introduced in Section 3.5.

3.1 An Overview of MG-HIF

Figure 1 exhibits the architecture of the proposed MG-HIF method. To solve the data sparsity and noise problem, this method is designed to extract heterogeneous interaction features from user–item bipartite graph and social relation network, which can provide complementary topological features for graph representation learning from different perspectives. Thus, the core idea of MG-HIF is to fuse heterogeneous interaction information from multi-graph to improve the vertex representation capability. Specifically, the MG-HIF model mainly consists of three components: (1) *a meta-path cross-fusion model* that encodes multi-hop neighborhood relations via meta-path from user–item bipartite graph via discrete cross-correlation to capture heterogeneous interaction information; (2) *a social relation GAN model* that learns the latent friendships for each user in a social network to combat the challenge of data sparsity and noise; and (3) *an attention mechanism-based multi-graph information fusion model* that fuses the outputs of two above models to enhance relation feature learning of vertex.

Meta-path Cross-Fusion Model. This model, at first, select all the user–item pairs $\{(u_i, v_j)\}$, $\forall u_i \in \mathcal{U}, \forall v_j \in \mathcal{V}$ from the user–item bipartite graph $\mathcal{G}_B$ and then generate meta-paths for each user–item pair (u_i, v_j) via random walk, namely $\mathscr{P}_{u_i,v_j} := RandomWalk(u_i, v_j)$, where the function $RandomWalk(u_i, v_j)$ denotes a path generated by random walk from u_i to v_j. Then, for different types of meta-path, a discrete cross-correlation is conducted on these meta-path pairs to implement multi-hop neighborhood relation feature cross-fusion. Compared with the existing techniques that only consider one-hop neighbors, this model fuses multi-hop heterogeneous interactions to improve the vertex feature learning.

Social relation GAN Model. This model leverages graph generative adversarial network technique, which aims to learn latent relation features from a given connectivity pattern distribution

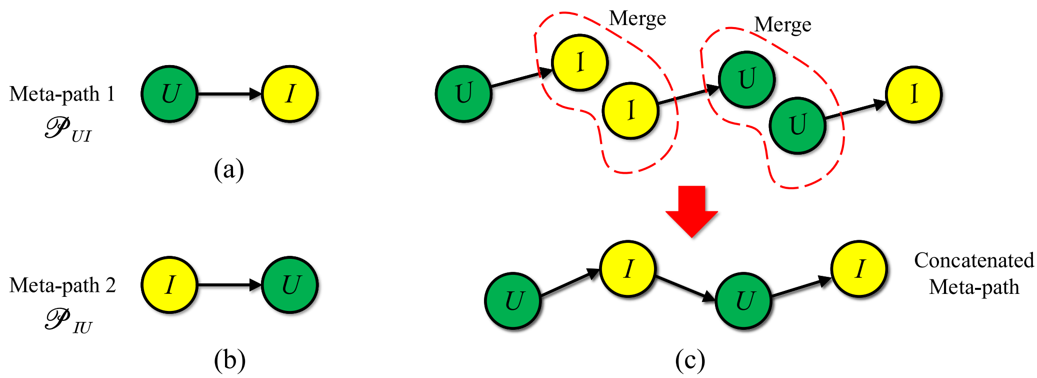


Fig. 3. The concatenation of meta-paths $\mathscr{P}_{IU}$ and $\mathscr{P}_{UI}$. According to the notions of meta-paths, the same types in different meta-paths can be merged. Best view in color.

$p_{true}(u|u_i)$ from a social relation network $\mathcal{G}_S$, where $u, u_i \in \mathcal{U}$. As introduced in Section 2.3, this model consists of two neural networks, i.e., a generator $G(\cdot|u_i;\boldsymbol{\theta}_G)$ and a discriminator $D(\hat{u}, u_i;\boldsymbol{\theta}_D)$. Via playing a minimax game, the generator $G(\cdot|u_i;\boldsymbol{\theta}_G)$ tries to fit the connectivity pattern distribution of the given graph $\mathcal{G}_S$ and then selects latent friendships of each user. The discriminator $D(u, u_i;\boldsymbol{\theta}_D)$ does its utmost to identify the selected neighbors and the original neighbors. Due to the latent connectivity features that are encoded into users' representations, the data sparsity and noise problem in the original social relation network can be alleviated.

Multi-graph Information Fusion Model. This model is built on the top of meta-path cross-fusion model and social relation GAN model, which receives the outputs of the above two components and fuses these heterogeneous interaction features together. Multi-graph information fusion can effectively improve the vertex representation, since the interaction features of user–item and user–user relations can complement each other to represent users' preferences more comprehensively and accurately. Besides, for both vertex representations of meta-path cross-fusion model and social relation GAN model, this model uses two attention modules (each for a type of output) to further enhance the representation.

After the multi-graph information fusion, the vertex representations are fed into a deep neural network $Dnn(\cdot;\boldsymbol{\theta}_{Dnn})$ to generate prediction.

3.2 Meta-path Cross-Fusion

For the interaction feature learning in user–item bipartite graph, a meta-path cross-fusion method is proposed to learn heterogeneous interaction feature of multi-hop neighbors to fuse more relation information than one-hop neighbors and then combat the challenge of early aggregation. In the following, we first introduce meta-path generation and then present the cross-fusion method.

Meta-path Generation. As discussed above, the user–item bipartite graph $\mathcal{G}_B$ contains two types of vertices $\mathcal{U}$ and $\mathcal{V}$, which are denoted as $\mathcal{A} = \{A_U, A_I\}$. Therefore, there are naturally two types of meta-paths in the bipartite graph $\mathcal{G}_B$, namely $\mathscr{P}_{UI} = (A_U, A_I)$, which is from a user vertex to an item vertex, and $\mathscr{P}_{IU} = (A_I, A_U)$, which is from an item vertex to a user vertex, as shown in Figure 3(a) and (b). $\mathscr{P}_{UI} = (A_U, A_I)$ represents a user interacts with an item, such as rating or buying. $\mathscr{P}_{IU} = (A_I, A_U)$ means an item is attracted by a user. According to the definition of meta-path presented in Section 2.2, two meta-paths can be concatenated by merging the same tail node and head node. Thus, we can generate compound meta-paths via concatenating $\mathscr{P}_{UI} = (A_U, A_I)$ and $\mathscr{P}_{IU} = (A_I, A_U)$, as shown in Figure 3(c). To consider more comprehensive

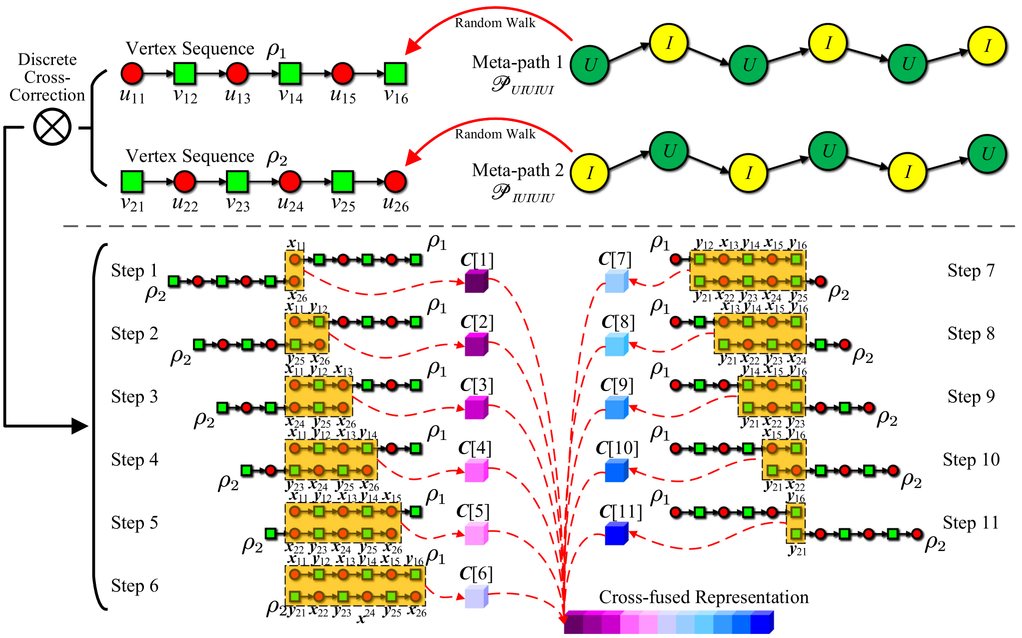


Fig. 4. An illustration of example of cross-fusion operation between two vertex sequences ρ_1 and ρ_2 generated from meta-path $\mathscr{P}_{UIUIUI}$ and $\mathscr{P}_{IUIUIU}$. After the vertex generation via random walk, a discrete cross-correlation is conducted on vertex sequence pairs to generate cross-fused representation. Best view in color.

connectivity patterns, we generate the compound meta-paths as follows:

$$\begin{aligned}
\mathscr{P}_{UI \bowtie IU \bowtie UI} &= (A_U, A_I, A_U, A_I), \\
\mathscr{P}_{UI \bowtie IU \bowtie UI \bowtie IU \bowtie UI} &= (A_U, A_I, A_U, A_I, A_U, A_I), \\
\mathscr{P}_{IU \bowtie UI \bowtie IU} &= (A_I, A_U, A_I, A_U), \\
\mathscr{P}_{IU \bowtie UI \bowtie IU \bowtie UI \bowtie IU} &= (A_I, A_U, A_I, A_U, A_I, A_U).
\end{aligned} \tag{6}$$

To simplify the expression, we denote $UI \bowtie IU \bowtie UI$ as $UIUI$ in the rest of the article. Consequently, including above four compound meta-paths, we get six meta-paths, i.e., $\mathscr{P}_{UI}$, $\mathscr{P}_{UIUI}$, $\mathscr{P}_{UIUIUI}$, $\mathscr{P}_{IU}$, $\mathscr{P}_{IUIU}$, $\mathscr{P}_{IUIUIU}$, as the template to generate vertex sequences that are corresponding to six connectivity patterns.

A random walk strategy is applied based on the aforementioned meta-paths to generate vertex sequences. The possibility distribution of random walk is

$$p(\mu_{i+1}|\mu_i) = \begin{cases} \frac{1}{|\mathcal{N}_{\mathscr{P}}(\mu_i)|} & , \forall \mu_i, \mu_{i+1} \in A_i, A_{i+1}, \\ 0 & , \text{otherwise}. \end{cases} \tag{7}$$

where μ_i denotes the ith vertex in the random walk, $\mathcal{N}_{\mathscr{P}}(\mu_i)$ is the set of one-hop neighbors of vertex μ_i along meta-path $\mathscr{P} = (A_I, A_U, \ldots, A_U)$ (or $\mathscr{P} = (A_U, A_I, \ldots, A_I)$). These vertex sequences generated randomly guided by meta-paths are the concrete representations of connectivity patterns of multi-hop neighbors. Learning interaction features from these vertex sequence can improve the vertex representation capability.

Cross-Fusion. After the meta-path and vertex sequence generation, we obtain the vertex sequence set $\mathcal{S} = \{\mathcal{S}_{UI}, \mathcal{S}_{UIUI}, \mathcal{S}_{UIUIUI}, \mathcal{S}_{IU}, \mathcal{S}_{IUIU}, \mathcal{S}_{IUIUIU}\}$, each subset in $\mathcal{S}$ represents the vertex sequences that are generated from the corresponding meta-path. For example, subset $\mathcal{S}_{UIUIUI}$ is generated from meta-path $\mathscr{P}_{UIUIUI}$. In each sequence, each vertex is replaced by its vertex representation vectors. To capture connectivity features for each user–item pair (u_i, v_j) comprehensively, a discrete cross-correlation operation is conducted on two vertex sequences with u_i or v_j as the head vertex, respectively. Cross-correlation function is widely used in the field of digital signal processing, which is to measure the degree of correlation of two signals, defined as follows: for two continuous signals $f(x)$ and $g(x)$,

$$\begin{aligned} \bar{\mathscr{R}}_{f,g}(m) &= f(x) \otimes g(x) \\ &= \int_{-\infty}^{+\infty} f^*(t) \times g(t+m)\, dt, \end{aligned} \tag{8}$$

for two discrete signals $f[x]$ and $g[x]$,

$$\begin{aligned} \ddddot{\mathscr{R}}_{f,g}(m) &= f(x) \otimes g(x) \\ &= \sum_{t=-\infty}^{+\infty} f^*[t] \times g[t+m], \end{aligned} \tag{9}$$

as the vertex sequences are the sequences of vertex representations, which can be seen as discrete signals. We employ discrete cross-correlation to measure the correlations between two vertex sequences to represent the multi-hop neighbors connectivity features. For example, as shown in Figure 4, vertex sequences of vertex u_{11} and v_{21} denoted as $\rho_1(u_{11}) = (u_{11}, v_{12}, u_{13}, v_{14}, u_{15}, v_{16}) \in \mathcal{S}_{UIUIUI}$ and $\rho_2(v_{21}) = (v_{21}, u_{22}, v_{23}, u_{24}, v_{25}, u_{26}) \in \mathcal{S}_{IUIUIU}$ are generated from meta-paths $\mathscr{P}_{UIUIUI}$ and $\mathscr{P}_{IUIUIU}$ via random walk. The embedding matrices of $\rho_1(u_{11})$ and $\rho_2(v_{21})$, denoted as $\boldsymbol{M}(\rho_1(u_{11})) = (\boldsymbol{x}_{11}, \boldsymbol{y}_{12}, \boldsymbol{x}_{13}, \boldsymbol{y}_{14}, \boldsymbol{x}_{15}, \boldsymbol{y}_{16})$ and $\boldsymbol{M}(\rho_2(v_{21})) = (\boldsymbol{x}_{21}, \boldsymbol{y}_{22}, \boldsymbol{x}_{23}, \boldsymbol{y}_{24}, \boldsymbol{x}_{25}, \boldsymbol{y}_{26})$, and the multi-hop neighbors connectivity representation matrix $\boldsymbol{C}\Big(\boldsymbol{M}(\rho_1(u_{11})), \boldsymbol{M}(\rho_2(v_{21}))\Big)$ can be calculated by discrete cross-correlation between $\boldsymbol{M}(\rho_1(u_{11}))$ and $\boldsymbol{M}(\rho_2(v_{21}))$ as follows:

$$\begin{aligned} \boldsymbol{C}\Big(\boldsymbol{M}(\rho_1(u_{11})), \boldsymbol{M}(\rho_2(v_{21}))\Big)[m] &= \boldsymbol{M}(\rho_1(u_{11})) \otimes \boldsymbol{M}(\rho_2(v_{21})) \\ &= \sum_{t=1}^{6} \boldsymbol{M}(\rho_1(u_{11}))[t] \times \boldsymbol{M}(\rho_2(v_{21}))[t+m], \end{aligned} \tag{10}$$

where $\boldsymbol{C}\Big(\boldsymbol{M}(\rho_1(u_{11})), \boldsymbol{M}(\rho_2(v_{21}))\Big)[m]$ denotes the mth vector of the matrix. For all the meta-paths of vertex u_{11} and v_{21}, the embedding matrices of these two vertices are denoted as $\mathbb{M}(\rho_\lambda(u_{11})) = \Big(\boldsymbol{M}(\rho_\lambda(u_{11}))\Big)\Big|_{\lambda=1}^{\Lambda_1}$ and $\mathbb{M}(\rho_\lambda(u_{21})) = \Big(\boldsymbol{M}(\rho_\lambda(u_{21}))\Big)\Big|_{\lambda=1}^{\Lambda_2}$, where Λ_1 and Λ_2 are the number of all sequences. Then the cross-fused representations $\mathbb{C}\Big(u_{11}, v_{21}\Big)$ between u_{11} and v_{21} are calculated as

$$\mathbb{C}\Big(u_{11}, v_{21}\Big) = \frac{1}{\Lambda_1 + \Lambda_2} \sum_{\lambda_1, \lambda_2 = 1}^{\Lambda_1, \Lambda_2} \boldsymbol{C}\Big(\boldsymbol{M}(\rho_{\lambda_1}(u_{11})), \boldsymbol{M}(\rho_{\lambda_2}(v_{21}))\Big). \tag{11}$$

As the large amount of computation of discrete cross-correlation, we propose to employ Fourier transform to reduce computational complexity. Specifically, embedding matrices of two vertex sequences $\rho_1(u_{11})$ and $\rho_2(v_{21})$ are projected into the Fourier domain, in which the discrete cross-correlation can be calculated by multiplication, and the results are projected into the original domain via inverse Fourier transform as follows:

$$\boldsymbol{C}\Big(\boldsymbol{M}(\rho_1(u_{11})), \boldsymbol{M}(\rho_2(v_{21}))\Big)[m] = \mathscr{F}^{-1}\Big(\mathscr{F}\Big(\boldsymbol{M}(\rho_1(u_{11}))\Big) \times \mathscr{F}\Big(\boldsymbol{M}(\rho_2(v_{21}))\Big)\Big). \tag{12}$$

By using FFT algorithm, the computational complexity can be reduced from $O(|\rho|^2)$ by discrete cross-correlation to $O(|\rho| log(|\rho|))$, where $|\rho|$ denotes the length of vertex sequence.

3.3 Social Relation GAN

To capture social relation features to support recommendation, we propose to learn user vertex embeddings that contain the social relationship information. Several techniques, e.g., GNN and GCN, introduced in the existing literature, are qualified for the graph embedding task. However, they cannot be used to solve the data sparsity problem in social networks. Therefore, in this work, we develop a social relation GAN model to find latent friendships of each user to diminish data sparsity and data noise problem. As introduced in Section 2.3, this model is a graphGAN-based network consisting of two components: a generator $G(\cdot|u_i;\boldsymbol{\theta}_G)$ to generate (or select) latent friends of a given user by fitting the connectivity pattern distribution in the social relation network $\mathcal{G}_S$ and a discriminator $D(\hat{u}_j, u_i;\boldsymbol{\theta}_D)$ to identify the outputs $\{\hat{u}_j\}$ of generator from the real samples $\{u_i\}$. They play a minimax game during the training to improve their capability step by step.

For the discriminator $D(\hat{u}_j, u_i;\boldsymbol{\theta}_D)$, we use a sigmoid function of the inner product of these two representations to implement the discrimination as follows:

$$D(\hat{u}_j, u_i;\boldsymbol{\theta}_D) = \sigma\left(\hat{\boldsymbol{x}}_j^\top \boldsymbol{x}_i\right) = \frac{1}{1 + exp(-\phi_D(\hat{u}_j;\boldsymbol{\theta}_D)^\top \phi_D(u_i;\boldsymbol{\theta}_D))}, \tag{13}$$

where $\phi_D(\cdot;\boldsymbol{\theta}_D)$ denotes a vertex embedding function in discriminator, $\hat{\boldsymbol{x}}_j = \phi_D(\hat{u}_j;\boldsymbol{\theta}_D)$ and $\boldsymbol{x}_i = \phi_D(u_i;\boldsymbol{\theta}_D)$. For the generator $G(\cdot|u_i;\boldsymbol{\theta}_G)$, inspired by Reference [54], we utilize graph softmax to implement it as follows:

$$G(\hat{u}_j|u_i;\boldsymbol{\theta}_G) = \left(\prod_{k=1}^{|\rho(u_i,\hat{u}_j)|} p_r\left(u_k^{\rho}\middle|u_{k-1}^{\rho}\right)\right) \times \left(p_r\left(u^{\rho}_{|\rho(u_i,\hat{u}_j)|-1}\middle|u^{\rho}_{|\rho(u_i,\hat{u}_j)|}\right)\right), \tag{14}$$

where $\rho(u_i, \hat{u}_j)$ is a path with the head vertex u_i and tail vertex $\hat{u}_j$. $|\rho(u_i, \hat{u}_j)|$ is the number of vertices in the path. u^{ρ} is the vertex in the path. Conditional probability $p_r(u_j|u_i)$ is the relevance probability of u_j *in* $\mathcal{N}_S(u_i)$ as

$$p_r(u_j|u_i) = \frac{exp(\phi_G(u_j;\boldsymbol{\theta}_G)^\top \phi_G(u_i;\boldsymbol{\theta}_G))}{\sum_{u_k \in \mathcal{N}(u_i)} exp(\phi_G(u_k;\boldsymbol{\theta}_G)^\top \phi_G(u_i;\boldsymbol{\theta}_G))}, \tag{15}$$

where $\phi_G(\cdot;\boldsymbol{\theta}_G)$ denotes a vertex embedding function in generator.

3.4 Multi-graph Information Fusion

After the vertex features are learned from user–item bipartite graph $\mathcal{G}_B$ and social relation network $\mathcal{G}_S$, these features are fused by an attention mechanism-based multi-graph information fusion model to improve the vertex representation learning. For the user–item graph, an item attention network is used to learn a set of weights for all the items that are connected to each user vector via meta-paths. For the social relation network, a user–user attention network is used to learn a set of weights for all the neighbors of each user. After the weighted features are produced, these two types of features are fused together via concatenation operation.

User-Item Attention Network. For each user $u_i \in \mathcal{U}$ and each item $v_j \in \mathcal{V}$, $\mathcal{N}_{\rho}^{\epsilon}(u_i, v_j)$ represents all the items and users that are in the vertex sequence under ϵth type of meta-path, where $\epsilon \in \{UI, UIUI, UIUIUI\} * \{IU, IUIU, IUIUIU\}$ denotes the interaction type of meta-path and the operator $*$ represents the Cartesian product. An attention network $Att_{UI}(\mathbb{C}(u_i, v_j);\boldsymbol{\theta}_{UI})$ learns a set of weights for all the items according to their representations and rating representation and then aggregates all these representations to produce this user's representation. This network can be formulated as

$$\xi_{ij}^{\epsilon} = \sigma\left(\boldsymbol{W}_{UI} \times \left(\sum_{\iota=1}^{L} \omega_{\iota} \times \sigma\left(\boldsymbol{W}_{R} \times \left(\boldsymbol{c}_{\iota} \bowtie \sum_{\kappa,\gamma=1}^{|\mathcal{N}_{\rho_{\iota}}^{\epsilon}(u_i,v_j)|} \boldsymbol{e}_{B}(u_{\kappa}, v_{\gamma})\right) + \boldsymbol{\beta}_{R}\right)\right) + \boldsymbol{\beta}_{UI}\right), \tag{16}$$

where σ denotes non-linear activation function; $\boldsymbol{c}_{\iota}$ is the ιth vector in matrix $\mathbb{C}(u_i, v_j)$, which is produced by ιth step of cross-correlation; L is the length of the vertex sequence; ι is the index of the cross-correlation step; and $\mathcal{N}_{\rho_{\iota}}^{\epsilon}(u_i, v_j)$ is the vertices that participate in the ιth step cross-correlation. $\boldsymbol{W}_{UI}$ and $\boldsymbol{\beta}_{UI}$ represent the parameters of user–item aggregation neural network. $\boldsymbol{W}_{R}$ and $\boldsymbol{\beta}_{R}$ represent the parameters of item-rating representation neural network. $\boldsymbol{e}_{B}(u_{\kappa}, v_{\gamma})$ denotes the rating representation between u_{κ} and v_{γ}. ω_{ι} denotes the weight of ιth vector, which is learned by the following two-layer attention network:

$$\hat{\omega}_{\iota} = \boldsymbol{W}_{2} \times \sigma\left(\boldsymbol{W}_{1} \times \sigma\left(\boldsymbol{W}_{R} \times \sum_{\kappa,\gamma=1}^{|\mathcal{N}_{\rho_{\iota}}^{\epsilon}(u_i,v_j)|} \left(\boldsymbol{y}_{\gamma} \bowtie \boldsymbol{e}_{B}(u_{\kappa}, v_{\gamma}) \bowtie \boldsymbol{x}_{\kappa}\right) + \boldsymbol{\beta}_{R}\right) + \boldsymbol{\beta}_{1}\right) + \beta_{2}, \tag{17}$$

$$\omega_{\iota} = \frac{exp(\hat{\omega}_{\iota})}{\sum_{\kappa,\gamma=1}^{|\mathcal{N}_{\rho}^{\epsilon}(u_i,v_j)|} exp(\hat{\omega}_{\iota})}. \tag{18}$$

Therefore, the representation of user–item pairs (u_i, v_j) with the attention mechanism under all the meta-paths is the concatenation of all these vectors and is calculated as follows:

$$\xi_{ij} = \biguplus_{\epsilon} \xi_{ij}^{\epsilon}, \tag{19}$$

where the $\biguplus$ is the concatenation operator.

User-User Attention Network. Similarly to the user–item attention network, a user–user attention network $Att_{UU}(;\boldsymbol{\theta}_{UU})$ is to learn a set of weights for all the neighbors for each user u_i in social relation network $\mathcal{G}_S$. This network can be formulated as

$$\zeta_i = \sigma\left(\boldsymbol{W}_{UU} \times \left(\sum_{\kappa=1}^{|\mathcal{N}_U(u_i)|} \varpi_{i\kappa} \times \boldsymbol{h}_{i\kappa}\right) + \boldsymbol{\beta}_{UU}\right), \tag{20}$$

where $\varpi_{i\kappa}$ represents the weight of κth neighbor of user u_i. Similarly to the above operation, $\varpi_{i\kappa}$ is produced by the following network:

$$\hat{\varpi}_{i\kappa} = \boldsymbol{W}_{2} \times \sigma\left(\boldsymbol{W}_{1} \times \sigma\left((\boldsymbol{h}_{i\kappa} \bowtie \boldsymbol{x}_{i}) + \boldsymbol{\beta}_{R}\right) + \boldsymbol{\beta}_{1}\right) + \beta_{2}, \tag{21}$$

$$\varpi_{i\kappa} = \frac{exp(\hat{\varpi}_{i\kappa})}{\sum_{\kappa=1}^{|\mathcal{N}_U(u_i)|} exp(\hat{\varpi}_{i\kappa})}. \tag{22}$$

Information Fusion Model. To improve the vertex representation learning, an information fusion model is used to fuse the representation ξ_{ij} from user–item bipartite graph $\mathcal{G}_B$ and the representation ζ_i from social relation network $\mathcal{G}_S$ together. This model is implemented by an l-layers deep neural network model $z_{ij} = Dnn(\xi_{ij}, \zeta_i; \boldsymbol{\theta}_{Dnn})$. Specifically, the first layer is a concatenation operator that concatenates ξ_{ij} and ζ_i, and the remaining several layers construct a multiple-layer

Table 2. Statistics of Evaluation Datasets

Property	Ciao	Filmtrust	Epinions	Douban
Number of Users	7,375	1,508	40,163	2,848
Number of Items	99,746	2,071	139,738	39,586
Number of Ratings	278,483	35,497	664,824	894,887
Density	0.0379%	1.1367%	0.0118%	0.7937%
Number of Relations	111,781	1,642	487,183	35,770

Perceptron. The structure of this DNN can be formulated as

$$\begin{aligned} \boldsymbol{a}_{ij}^{(1)} &= \boldsymbol{\xi}_{ij} \bowtie \boldsymbol{\zeta}_i, \\ \boldsymbol{a}_{ij}^{(2)} &= \sigma\left(\boldsymbol{W}_2 \times \boldsymbol{a}_{ij}^{(1)} + \boldsymbol{\beta}_2\right), \\ &\cdots \\ \boldsymbol{z}_{ij} &= \sigma\left(\boldsymbol{W}_l \times \boldsymbol{a}_{ij}^{(l)} + \boldsymbol{\beta}_l\right), \end{aligned} \tag{23}$$

where $\boldsymbol{a}^{(j)}$ is the output of jth layer, $j \in [1, \ldots, l]$. The parameter vector is $\boldsymbol{\theta}_{Dnn} = (\{\boldsymbol{W}_j\}|_{j=2}^{l}, \{\boldsymbol{\beta}_j\}|_{j=2}^{l})$. Then the fused representations are fed into a linear prediction layer to generate the predicted recommendation as follows:

$$\tau'_{ij} = Predict(\boldsymbol{z}_{ij}; \boldsymbol{\theta}_P) = \boldsymbol{W} \times \boldsymbol{z}_{ij}. \tag{24}$$

3.5 Optimization

To learn the rating prediction model, we utilize the following loss function to implement the training:

$$\mathcal{L}_{rating}(\boldsymbol{\theta}_{UI}, \boldsymbol{\theta}_{UU}, \boldsymbol{\theta}_{Dnn}, \boldsymbol{\theta}_P) = \frac{1}{2|\mathcal{E}_B|} \sum\nolimits_{i,j \in \mathcal{E}_B} (\tau'_{ij} - \tau_{ij})^2. \tag{25}$$

The loss function of the social relation GAN is

$$\mathcal{L}(\boldsymbol{\theta}_G, \boldsymbol{\theta}_D) = \sum_{i=1}^{|\mathcal{U}|} \left(\mathbb{E}_{u \sim p_{true}(\cdot|u_i)} \left[logD(\hat{u}_j, u_i; \boldsymbol{\theta}_D)\right] + \mathbb{E}_{u \sim p_G(\cdot|u_i)} \left[log(1 - D(\hat{u}_j, v_i; \boldsymbol{\theta}_D))\right]\right). \tag{26}$$

Therefore, the total objective function can be formulated as

$$\begin{aligned} \left(\boldsymbol{\theta}_G^*, \boldsymbol{\theta}_{UI}^*, \boldsymbol{\theta}_{UU}^*, \boldsymbol{\theta}_{Dnn}^*, \boldsymbol{\theta}_P^*\right) &= \arg \min_{\boldsymbol{\theta}_G, \boldsymbol{\theta}_{UI}, \boldsymbol{\theta}_{UU}, \boldsymbol{\theta}_{Dnn}, \boldsymbol{\theta}_P} \left(\mathcal{L}_G(\boldsymbol{\theta}_G) + \mathcal{L}_{rating}(\boldsymbol{\theta}_{UI}, \boldsymbol{\theta}_{UU}, \boldsymbol{\theta}_{Dnn}, \boldsymbol{\theta}_P)\right), \\ \boldsymbol{\theta}_D^* &= \arg \max_{\boldsymbol{\theta}_D} \mathcal{L}_D(\boldsymbol{\theta}_D). \end{aligned} \tag{27}$$

4 EXPERIMENTS

To evaluate the effectiveness of our proposed model, we conduct a set of the experiments on four public benchmark social recommendation datasets. In the following, we first introduce our four public benchmark datasets (Ciao, Epinions, Filmtrust, Douban, and evaluation metrics) and then discuss the implementation details. Finally, we compare our proposed method with seven state-of-the-art techniques and analyze the experiment results.

4.1 Experimental Settings

4.1.1 Dataset Description. We conduct the comprehensive experiments on four public benchmark datasets: Ciao, Epinions, Filmtrust, and Douban; some important properties of them are summarized in Table 2. A brief introduction of them is as follows:

- **Ciao**.[1] Ciao is a product review site, where users can evaluate products by writing comments and build a trusted friends' network with the users shared the common interests. The Ciao dataset consists of two parts: user–item interaction and directed trust social relationships. The items in user–item interaction are rated from 1 (min) to 5 (max).
- **Epinions**.[2] Epinions is a who-trust-whom online social network of a general consumer review site, which was collected by Paolo Messa et al. from famous online shopping website epinions. In this website, users can review items, assign them with an integer ranging from 1 to 5, and set up trust relationships with others to facilitate making decisions.
- **FilmTrust**.[3] FilmTrust is a small dataset crawled from the entire FilmTrust website in June, 2011. Similarly to Ciao, it contains user–item interaction information and social relationships by active users. Different as Ciao, FilmTrust employs 4-star system to rate items. FilmTrust has eight different ratings from 0.5 to 4.0 with step 0.5.
- **Douban**.[4] Douban is taken from anonymous comments of popular social networking website Douban(https://www.douban.com/). Similarly to Ciao and Epinions, Douban employs 5-star system to rate items. On average, users of Douban have 12.56 trust relations and 314.26 ratings.

4.1.2 Evaluation Metrics. We evaluate the compared methods by using two classical metrics **Mean Absolute Error (MAE)** and **Root Mean Squared Error (RMSE)** on all the four datasets. MAE and RMSE metrics are widely used in recommendation system to evaluate the closeness of predicted ratings to the true ratings.

The metric MAE is defined as

$$MAE = \frac{1}{|\mathcal{E}|} \sum_{(u_i, v_j) \in \mathcal{E}} |\tau'_{ij} - \tau_{ij}|, \tag{28}$$

where $|\mathcal{E}|$ is the size of $\mathcal{E}$, τ'_{ij} is the predicted rating from user u_i to item v_j, and τ_{ij} is the true rating from user u_i to item v_j.

The metric RMSE is defined as

$$RMSE = \sqrt{\frac{\sum_{(u_i, v_j) \in \mathcal{E}} (\tau'_{ij} - \tau_{ij})^2}{|\mathcal{E}|}}. \tag{29}$$

A smaller MAE and RMSE value means better prediction accuracy. Note that Koren [31] claims that small improvements in MAE and RMSE can have a significant impact on the quality of the top-few recommendation.

4.1.3 Competitors. To verify the effectiveness of our proposed technique, we compare our proposed model with seven comparative methods, including some state-of-the-art techniques for recommendation and social recommendation. The recommendation models only include Slope-one [34], this model leverages user–item interactions as model input. The social recommendation models include three traditional methods, namely SoRec [42], RSTE [41], and SocialMF [26], and three deep learning models, namely SREPS [38], DeepSoR [12], and DANSER [63]. Both user–item interactions and user social network information are fully utilized in model construction. The brief introductions of these competitors are shown as follows.

[1] http://www.cse.msu.edu/~tangjili/index.html.
[2] http://www.trustlet.org/wiki/Epinions_datasets.
[3] http://trust.mindswap.org/ont/trust.owl.
[4] http://smiles.xjtu.edu.cn/Download/download_Douban.html.

Table 3. Performance Comparisons on Ciao Dataset

	Ciao (70%)		Ciao (80%)		Ciao (90%)	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
Slopeone	0.7926	1.0914	0.7854	1.0848	0.7814	1.0728
SoRec	0.7798	1.0375	0.7677	1.0211	0.759	1.0126
RSTE	0.7891	1.0709	0.7824	1.0666	0.7792	1.0603
SocialMF	0.7814	1.055	0.7728	1.0516	0.7622	1.0484
DANSER	0.7367	0.9877	0.7332	0.9737	0.7324	0.9656
DeepSoR	0.7376	1.0064	0.7265	0.9888	0.7183	0.9791
SREPS	0.7493	1.0132	0.7418	1.0044	0.7336	0.9964
Ours	**0.7246**	**0.9836**	**0.7151**	**0.9701**	**0.7084**	**0.9623**

- **Slopeone** [34] is an online rating-Based collaborative filtering that exploits the score difference between different items to predict the user's rating of items.
- **SoRec** [42] proposes a social recommendation method integrating a user–item rating matrix with the user's social network structure by using probabilistic matrix factorization.
- **RSTE** [41] adopts a novel probabilistic factor analysis framework, which utilizes social trust ensemble to represent the formulation of the social trust restrictions, to fuse the users' interests and trusted friends' favors.
- **SocialMF** [26] explores trust information and its propagation to incorporate social information into the basic matrix factorization model to enhance quality of traditional recommendations.
- **SREPS** [38] proposes a novel method to distinguish the differences between user preference in recommendation system and that in social networks.
- **DeepSoR** [12] presents a DNN-based model to find non-linear features of each user from social relations and to integrate into probabilistic matrix factorization for social rating prediction problem.
- **DANSER** [63] investigates dual graph attention networks to collaboratively learn representations for twofold social effects and extends the social effects in user domain to item domain.

4.1.4 Implementation Details. We implemented our proposed method on a well-known Python library TensorFlow and deployed our model on GTX 1080ti GPU with 11G memory, Intel Core i7-6800K CPU @3.4 GHz. The hyper-parameter settings for each dataset are as follows: input batch size 256, embedding dimension 64, and learning rate 0.001. We split the user–item pair in bipartite graph into training set (α%) and testing set (1- α%), where α was varied as 70%, 80%, and 90%. Half of the testing set is used as a validation set to tune hyper-parameters, and the other half is used for the final performance comparison. The default setting of α is 80%. For mate-path length, we varies it from 2 to 6 with length 2, the default setting of mate-path length is 4.

4.2 Performance Comparisons on Different Datasets

Comparisons on Ciao. We first compare the performance of all competitors in Ciao dataset. The overall MAE and RMSE values among all competitors is shown in Table 3. We have the following findings:

- Compared with Slopeone, SoRec, RSTE, and SocialMF always have better performance. SoRec, RSTE, and SocialMF employ both user–item interactions information and social network information, while Slopeone only takes the user–item interaction information into

Table 4. Performance Comparisons on Epinions Dataset

	Epinions (70%)		Epinions (80%)		Epinions (90%)	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
Slopeone	0.9152	1.2279	0.8994	1.2124	0.8963	1.2074
SoRec	0.8877	1.1698	0.8819	1.1616	0.8682	1.148
RSTE	0.8905	1.1955	0.8824	1.1834	0.8739	1.1804
SocialMF	0.8914	1.1919	0.8847	1.1815	0.8724	1.1669
DANSER	0.7986	1.0479	0.7933	1.0359	0.7808	1.0234
DeepSoR	0.8349	1.1452	0.8276	1.1035	0.8152	1.1179
SREPS	0.8265	1.0746	0.8171	1.0483	0.8065	1.0373
Ours	**0.7899**	**1.0344**	**0.7854**	**1.0236**	**0.7746**	**1.0143**

to rating prediction. This indicates that social network information does help improve the accuracy of rating predictions.

- DANSER, SREPS, and DeepSoR perform better than the traditional recommendation methods, such as SoRec, RSTE, and SocialMF. Although all of them take both user–item interactions information and social network information into consideration, the deep learning-based methods can further improve the quality of rating prediction by deep neural network models.
- Our proposed model MG-HIF beats all the comparison algorithms. Compared to the deep learning-based methods, our method employs a meta-path cross-fusion model to fuse multi-hop heterogeneous interaction features and a social relation GAN to explore latent friendships of each user to further improve the quality of rating prediction.

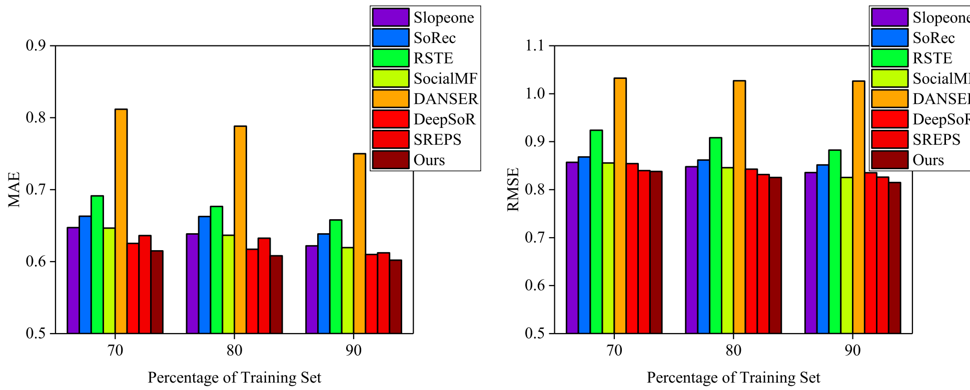
Comparisons on Epinions. The result of different percentages of training set on Epinions are shown in Table 4. Compared to the Ciao, Epinions has more users, items, and ratings information. Thus, the disadvantages of matrix-based decomposition models such as RSTE, SoRec, and SocialMF are gradually revealed, although neural network-based social recommendation methods like DeepSoR, DANSER, and SREPS perform moderately on smaller datasets. As the dataset size increases, the advantage of neural network becomes more and more obvious.

Similarly to Ciao, SoRec, RSTE, and SocialMF always perform better than Slopeone. Because Slopeone only considers the rating information, SoRec, RSTE, and SocialMF take both rating and social information into consideration. The algorithms based on neural networks, such as DeepSoR, DANSER, and SREPS, have better performance than the traditional algorithms based on matrix decomposition.

Among them, DANSER has the best performance. But the performance of our approach is still better than DANSER. Although both of these methods use graph network to integrate social node information, our method can learn the topological information of the user–item interaction network via meta-path and social relation GAN.

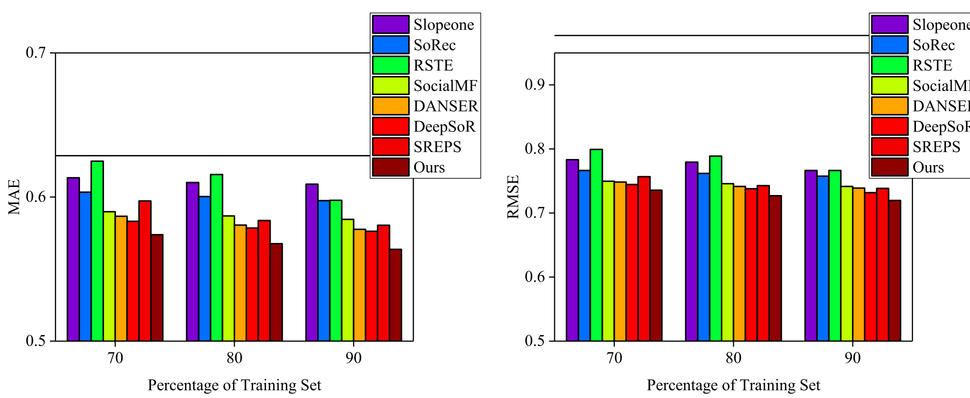
Comparisons on FilmTrust. Figure 5 shows the prediction errors of MAE and RMSE of all recommendation algorithms in FilmTrust. Compared with other datasets, FilmTrust has the smallest data size in terms of users, items, and ratings. The advantages of the traditional social recommendation algorithm based on the matrix decomposition model have been enhanced and achieved good performance.

Because the dataset size of Filmtrust is too small, the learning and representation ability of the neural network is seriously restricted, and the algorithm based on deep neural network cannot be well represented. In particular, the performance of DANSER, which is based on various complex neural network, degrades significantly.



(a) MAE (b) RMSE

Fig. 5. The MAE and RMSE of our method on FilmTrust dataset.



(a) MAE (b) RMSE

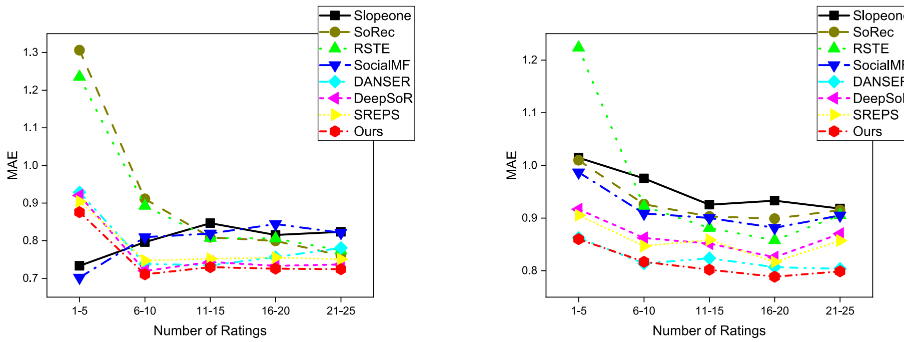
Fig. 6. The MAE and RMSE of our method on Douban dataset.

Our proposed method generates a large number of mate-paths that can make full use of the FilmTrust network to fuse multi-hop neighbor information. At the same time, social relation GAN is used to generate latent friendship of each user, which reduces the sparsity of data and alleviates the impact caused by small size of dataset.

Comparisons on Douban. We compare the performance of all competitors in the Douban dataset. The MAE and RMSE value among all methods are shown in Figure 6(a) and Figure 6(b), respectively. Obviously, compared with the performance on the FilmTrust dataset, both MAE and RMSE among all these competitors are lower than the former. That is because the Douban dataset is much larger than the FilmTrust dataset. On a larger dataset, more effective interaction information can be aggregated to improve the performance of prediction. However, the prediction of deep learning-based approaches is more precise than the traditional techniques. It is easy to understand that a larger dataset can provide more training samples for deep models to obtain more powerful representation capability.

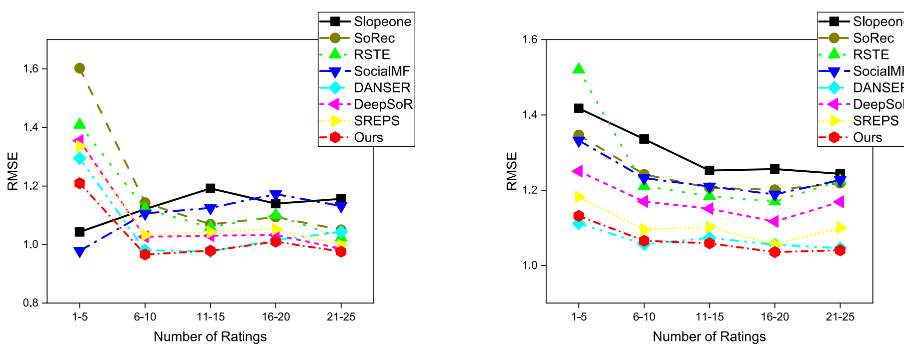
4.3 Performance Comparisons on Different Data Sparsity

To evaluate the capacity of MG-HIF in alleviating the data sparsity problem, we compared it with the above algorithms by simulating different levels of rating per user profile. More specifically,



(a) Ciao (b) Epinions

Fig. 7. Comparisons of MAE on Ciao and Epinions dataset.



(a) Ciao (b) Epinions

Fig. 8. Comparisons of RMSE on Ciao and Epinions dataset.

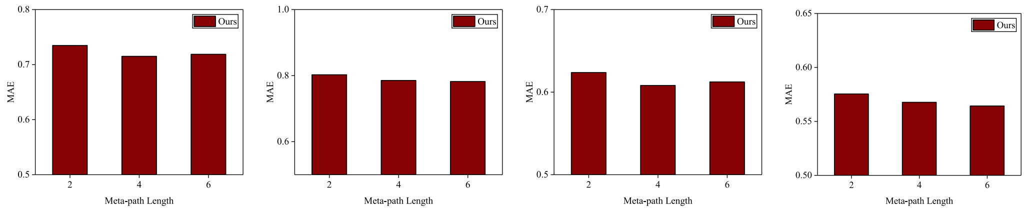
after obtaining the predicted results from the experiment, we disposed these results by selecting users whose number of ratings are located in specific ranges, (i.e., 1–5, . . . , 20–25).

Varying Number of Ratings in Terms of MAE. Figure 7 illustrates the MAE value of our method varies specific ranges from 1–5 to 20–25 on Ciao and Epinions. In most cases, MG-HIF outperforms comparison algorithms on both datasets, and the performance of deep learning algorithms are better than those of traditional recommendation methods. In Ciao, the performance of Slopeone and SocialMF is better than that of the deep learning algorithms when the user has a few ratings (less than 5), while the deep learning algorithms outperform Slopeone and SocialMF when the user has more ratings (more than 6). The possible reason for this phenomenon may be that the number of users participating in the statistics is too small. In Ciao, the number of users, whose rating number is less than 5, is less 50. In Epinions, the performance of MG-HIF is similar to that of Danser when the user has a few ratings (less than 10), while the MG-HIF is better than Danser when the user has more ratings (more than 11).

Varying Number of Ratings in Terms of RMSE. Figure 8 illustrates the RMSE value of our method varies specific ranges from 1–5 to 20–25 on Ciao and Epinions. Similarly to the trend of MAE, in Ciao, MG-HIF achieves the best performance when the user has more ratings (more than 6). MG-HIF outperforms Danser when the user has more ratings (more than 11) in Epinions.

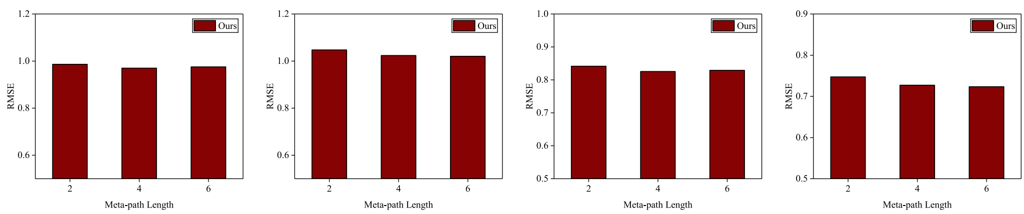
4.4 Performance Comparisons on Different Meta-path Length

MAE of Our Method Varying Meta-Path. Figure 9 illustrates that the MAE value of our method varies when the meta-path length increases from 2 to 6 on the Ciao, Epinions, FilmTrust, and



(a) Ciao (b) Epinions (c) FilmTrust (d) Douban

Fig. 9. The MAE of our method varying meta-path length on four datasets.



(a) Ciao (b) Epinions (c) FilmTrust (d) Douban

Fig. 10. The RMSE of our method varying meta-path length on four datasets.

Douban datasets. It is clear that with the increase of meta-path length from 2 to 4, the performance of our method is improved on these four datasets. More concretely, on the one hand, the changing of MAE on Ciao (Figure 9(a)) and FilmTrust (Figure 9(c)) from 2 to 4 is obvious, which can indicate that the longer the meta-path, more interaction information can be aggregated to enhance prediction. However, when the length of meta-path is increased to 6, the performance of our method has a slight decrease. Because the long meta-path may bring data noise into the vertex representation. On the other hand, with the other two large datasets, Epinions (Figure 9(c)) and Douban (Figure 9(d)), the performance of our method is best when the meta-path length is increased to 6. The main reason for this phenomenon is that the longer meta-path can aggregate more effective interaction information than noise on larger graphs.

RMSE of Our Method Varying Meta-Path. Figure 10 demonstrates that the RMSE value of our MG-HIF method varies when the meta-path length increases from 2 to 6 on the Ciao, Epinions, FilmTrust, and Douban datasets. Similarly to the changing of MAE, on Ciao and FilmTrust, MG-HIF gets the highest precision of prediction when the meta-path length is 2. However, while the length of meta-path equals 4, the performance of this setting is similar to the length that equals 6 for the Epinions and Douban datasets.

4.5 Limitations

According to Reference [52], different evaluation metrics are proposed to evaluate the quality of recommendation from different perspectives, such as prediction accuracy, ranking accuracy, diversity, and coverage. Recently, optimization for recommender systems has mainly been divided into rating prediction and Top-N recommendation, thus leading to two main branches of evaluation metrics: predication-aware error metrics (e.g., RMSE and MAE) and rank-aware accuracy metrics (e.g., NDCG@N, HR, and Pr@N, Re@N). In fact, most of the recent works in the field of recommender system is based on Top-N recommendation, but there are still some works only based on rating prediction, whose goal is to measure the closeness of predicted ratings to the true ratings, such as in References [12, 38, 61]. The goal of this article is not Top-N recommendation but rating prediction, so we use RMSE and MAE as the evaluation metrics. According to the conclusion drawn from Reference [7], there is no monotonic relation between error metrics and accuracy metrics. It means that even though a method achieves a rank accuracy for Top-N recommendation, it does not necessarily mean it will outperform other algorithms for the rating prediction.

To further verify the performance of the comparison algorithms of the current experiment under Top-N recommendation, we use the rank-aware accuracy metrics to evaluate the output of these comparison algorithms. The evaluation results of these comparison algorithms with rank-aware accuracy metrics differ from those with error metrics. In the case of adopting rank-aware accuracy metrics as the evaluation criteria, our model (MG-HIF) is not the best among all comparison algorithms. The main reasons are as follows: First, the modeling method adopted by the model differs. In the rating prediction scenario, the model uses a pointwise approach to model user preferences, that is, considers the degree of preference of a single user for a single item. In Top-N recommendation, the model uses a pairwise/listwise approach to model the user's preference ranking, that is, the model takes into account each user's preference ranking relationship for items. Second, the loss function used by the model differs. Top-N recommendation mainly uses the loss function based on ranking, such as Bayesian Personalized Ranking.

5 RELATED WORK

In this section, we briefly review the previous researches concerning to social recommendation, adversarial learning-based recommendation, and meta-path-based recommendation, which are related to our proposed approach.

5.1 Social Recommendation

Social recommendation [52, 71] has become a hot issue, attracting the attention of researchers, and is based on the homophily of social relationships [43]. In a nutshell, the behavior or preference of a user will always be influenced by his/her friends in an online social network, which is used by Wang et al. [55] to support image search on social media websites. According to the techniques used, the existing studies can be divided into two categories: matrix factorization-based methods and GNN-based methods, which are reviewed, respectively, in the following.

Matrix Factorization-based Methods. Matrix factorization is a conventional technique used in collaborative filtering and social recommendation [32, 32, 65] that evolved from the **singular value decomposition (SVD)** algorithm. Ma et al. [42] proposed a probabilistic matrix factorization-based [47] method to learn low-rank user latent feature space and item latent feature space via considering both social relations of users and rating records simultaneously. Jamali et al. [26] used the mechanism of trust propagation in matrix factorization to reduce recommendation error. Purushotham et al. [44] combined topic model and probabilistic matrix factorization of social networks to infer useful latent topics and social information for collaborative filtering. Liu et al. [39] proposed a Bayesian probabilistic matrix factorization–based method that fuses item contents information and user social information to improve performance. Chen et al. [3] developed a hierarchical Bayesian model that not only considers a topic model to exploit item content but also social matrix factorization to handle ratings and social relationships. Forsati et al. [16] utilized both trust and distrust relationships to enhance a matrix factorization-based approach to address the data sparsity and cold-start problem. Silva [8] proposed a Poisson matrix factorization with content and social trust information method that factorizes a non-negative user–item interaction matrix and a non-negative item–content matrix.

GNN-based Methods. To represent non-linear relation information of nodes, GNN [19, 72] is used in many recent studies [14] to improve the recommendation performance. Fan et al. [14] proposed a Deep Social Collaborative Filtering approach that considers the social network of users with various aspects to explore distant social relation information via GNN. Wu et al. [63] used a dual graph attention network to learn users' and items' representations from static and dynamic relations in both user and item domain. Song et al. [49] developed a dynamic-graph-attention neural network that employs RNN to model user behaviors in online communities to learn context-dependent social influence. Zhang et al. [69] proposed a Stacked and Reconstructed Graph Convolutional Networks method to enhance the node representation learning. In this technique, a stack of GCN-based encoder-decoders is used to generate latent factors of users and items in both transductive and inductive ways. Kim et al. [29] introduced a Heterogeneous Graph Propagation method that employs a personalized PageRank-based propagation scheme for both group-user graph and user–item graph to address the over-smoothing of GNNs. Xiao et al. [64] proposed a GNN-based technique to implement a mutualistic mechanism to learn the mutual reinforcement relationship between consumption behaviors and social behaviors. Wang et al. [58] employed Dual Graph Attention Networks to implement a metric learning-based model, which learns the relation vectors by neighborhood interactions. Vijaikumar et al. [53] exploits multi-head and multi-layer graph attention mechanism to construct a method named SoRecGAT, in which the influence of entities on each other can be learned more accurately via attention model. To deal with the problem that high-order neighbors increase dramatically with the order size, Liu et al. [40] proposes a graph neural network-based social recommender to learning better user embeddings.

5.2 Adversarial Learning-based Recommendation

Adversarial learning techniques, especially generative adversarial networks [18], have been applied in many areas, such as visual recognition [60], image generation [1], cross-modal retrieval [59], as well as graph-based recommendation [17]. To combat the problem of data sparsity and noise, graph-based GAN is used to explore latent neighbors or close relationships to improve the predictions or train multiple modules simultaneously in an adversarial manner. He et al. [21] developed an Adversarial Personalized Ranking model to boost the pairwise ranking method BPR via adversarial learning. Krishnan et al. [33] considered social influence and latent interests simultaneously and proposed an adversarial learning-based modular architecture-agnostic framework, which decouples the architectural choices for recommendation and social representation. Rafailidis et al. [46] studied the influence by users' reviews for recommendation via an adversarial training approach. They proposed to factorize the rating matrix by regularizing the representations of reviews with the user and item latent factors. With the help of adversarial learning, Fan et al. [11] designed a social recommendation approach named DASO to transfer users' information between the social domain and item domain. Yu et al. [66] presented an end-to-end GAN-based social recommendation method in which the generator is used to produce latent friends and the discriminator is to assess these generated friends and rank the items. Wu et al. [62] developed a Personalized Diversity promoting GAN to capture co-occurrence of diverse items via a Determinantal Point Process kernel matrix. Yu et al. [67] introduced a GCN-based adversarial learning approach, in which a GCN-based autoencoder is used to encode high-order and complex connectivity patterns to dismiss relation sparsity and noises and an attentive recommendation module to learn the heterogeneous strengths of social relations. Recently, Deldjoo et al. [9] published a comprehensive survey that introduces recent progress on adversarial recommendation systems and displays the achievement of adversarial machine learning in GANs for generative applications.

5.3 Meta-path-based Recommendation

Meta-path is a core notion of heterogeneous network analysis proposed in Reference [51], which is a path consists of several relations between different types of vertices. Burke et al. [2] proposed a social web recommendation technique that is improved by meta-path. According to meta-path, Shi et al. [48] developed a semantic path-based personalized recommendation method, SemRec, to predict the items recommendation. Liang et al. [36] studied HIN-based service recommendation problem and designed a method called PaSRec for mashup creation via meta-path-based similarity measurement. Hu et al. [23] exploited a deep co-attention mechanism and a priority-based sampling technique to learn high-quality meta-path representation. For music recommendation task, Fang et al. [15] developed a meta-paths-based method to learn user preferences via a combination of Bayesian Personalized Ranking model and representation learning in heterogeneous music networks. Song et al. [50] used meta-path to find out latent friends who share similar interests with active users in a heterogeneous network to enhance the performance of social recommender systems. Based on the idea of Funk-SVD, Ling et al. [37] proposed a time-weighted meta-path-based recommendation method with a time deviation matrix to predict the ratings. Fan et al. [10] presented a Meta-path-guided Embedding method for intent recommendation approach, in which a meta-path-guided heterogeneous graph neural network is employed to learn the embeddings of objects. Zhu et al. [73] proposed to utilize users' feedback to improve the recommendation and developed a Meta-Path and Adapted Attention-GRU-based method, which predicts the recommendation via the closely related feedbacks. Huang et al. [25] used meta-path from a heterogeneous information network and designed a position-based self-attention mechanism to learn local preference representation.

6 CONCLUSION

This article proposes a novel social recommendation approach, named MG-HIF, which combines the information from user–item graph and social relation network to improve the performance of recommendation. To combat the challenge of early summarization, we propose to utilize meta-path as a powerful tool for user–item bipartite graph, which represents the interaction features of multi-hop neighbors via discrete cross-correlation between different vertex sequences produced by different meta-paths, and then fuse these interaction features of user–item pairs to represent connectivity patterns comprehensively. To address the data sparsity and data noise problem, we propose an adversarial learning model named social relation GAN to explore latent friendships of each user from the social relation network, which is to provide richer social relationship information for recommendation prediction. Moreover, to further boost the interaction representation learning, we propose to fuse the representation from the user–item graph and social relation network via a novel multi-graph information fusion model with attention mechanism. To evaluate the performance of MG-HIF, we conduct comprehensive experiments on four frequently used datasets. Experimental results demonstrate that our method has better performance than states of the art.

REFERENCES

[1] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. 2017. CVAE-GAN: Fine-grained image generation through asymmetric training. In *Proceedings of the IEEE International Conference on Computer Vision*. 2745–2754.

[2] Robin D. Burke and Fatemeh Vahedian. 2013. Social web recommendation using metapaths. In *Proceedings of the 5th ACM RecSys Workshop on Recommender Systems and the Social Web Co-located with the 7th ACM Conference on Recommender Systems (RecSys'13)*, Bamshad Mobasher, Dietmar Jannach, Werner Geyer, Jill Freyne, Andreas Hotho, Sarabjot Singh Anand, and Ido Guy (Eds.), Vol. 1066. ACM, New York, NY.

[3] Chaochao Chen, Xiaolin Zheng, Yan Wang, Fuxing Hong, Zhen Lin, et al. 2014. Context-aware collaborative topic regression with social matrix factorization for recommender systems. In *Proceedings of the AAAI Annual Conference on Artificial Intelligence (AAAI'14)*, Vol. 14. 9–15.

[4] Yifan Chen, Pengjie Ren, Yang Wang, and Maarten de Rijke. 2019. Bayesian personalized feature interaction selection for factorization machines. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*, Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.). ACM, 665–674. https://doi.org/10.1145/3331184.3331196

[5] Yifan Chen, Yang Wang, Xiang Zhao, and Maarten De Rijke. 2020. Block-aware item similarity models for top-N recommendation. *ACM Trans. Inf. Syst.* 38, 4 (2020), 42:1–42:26. https://doi.org/10.1145/3411754.
[6] Yifan Chen, Yang Wang, Xiang Zhao, Hongzhi Yin, and Maarten De Rijke. 2020. Local variational feature-based similarity models for recommending top-n new items. *ACM Trans. Inf. Syst.* 38, 2 (2020), 12:1–12:33.
[7] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the 2010 ACM Conference on Recommender Systems, (RecSys'10)*, Xavier Amatriain, Marc Torrens, Paul Resnick, and Markus Zanker (Eds.). ACM, 39–46.
[8] Eliezer de Souza da Silva, Helge Langseth, and Heri Ramampiaro. 2017. Content-based social recommendation with poisson matrix factorization. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases ECML PKDD'17)* Lecture Notes in Computer Science, Michelangelo Ceci, Jaakko Hollmén, Ljupco Todorovski, Celine Vens, and Saso Dzeroski (Eds.), Vol. 10534. Springer, 530–546. https://doi.org/10.1007/978-3-319-71249-9_32
[9] Yashar Deldjoo, Tommaso Di Noia, and Felice Antonio Merra. 2021. A survey on adversarial recommender systems: from attack/defense strategies to generative adversarial networks. *ACM Comput. Surv.* 54, 2 (2021), 1–38.
[10] Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, and Yongliang Li. 2019. Metapath-guided heterogeneous graph neural network for intent recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'19)*, Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis (Eds.). ACM, 2478–2486. https://doi.org/10.1145/3292500.3330673
[11] Wenqi Fan, Tyler Derr, Yao Ma, Jianping Wang, Jiliang Tang, and Qing Li. 2019. Deep adversarial social recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19)*, Sarit Kraus (Ed.). ijcai.org, 1351–1357. https://doi.org/10.24963/ijcai.2019/187
[12] Wenqi Fan, Qing Li, and Min Cheng. 2018. Deep modeling of social relations for recommendation. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.). AAAI Press, 8075–8076.
[13] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *Proceedings of the World Wide Web Conference.* 417–426.
[14] Wenqi Fan, Yao Ma, Dawei Yin, Jianping Wang, Jiliang Tang, and Qing Li. 2019. Deep social collaborative filtering. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys'19)*, Toine Bogers, Alan Said, Peter Brusilovsky, and Domonkos Tikk (Eds.). ACM, 305–313. https://doi.org/10.1145/3298689.3347011
[15] Qianqi Fang, Ling Liu, Junliang Yu, and Junhao Wen. 2018. Meta-path based heterogeneous graph embedding for music recommendation. In *Proceedings of the 25th International Conference on Neural Information Processing (ICONIP'18),* Lecture Notes in Computer Science, Long Cheng, Andrew Chi-Sing Leung, and Seiichi Ozawa (Eds.), Vol. 11303. Springer, 101–113. https://doi.org/10.1007/978-3-030-04182-3_10
[16] Rana Forsati, Mehrdad Mahdavi, Mehrnoush Shamsfard, and Mohamed Sarwat. 2014. Matrix factorization with explicit trust and distrust side information for improved social recommendation. *ACM Trans. Inf. Syst.* 32, 4 (2014), 1–38.
[17] Min Gao, Junwei Zhang, Junliang Yu, Jundong Li, Junhao Wen, and Qingyu Xiong. 2021. Recommender systems based on generative adversarial networks: A problem-driven perspective. *Inf. Sci.* 546 (2021), 1166–1185.
[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems.* 2672–2680.
[19] Marco Gori, Gabriele Monfardini, and Franco Scarselli. 2005. A new model for learning in graph domains. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, Vol. 2. IEEE, 729–734.
[20] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. 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.* 639–648.
[21] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. 2018. Adversarial personalized ranking for recommendation. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'18)*, Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz (Eds.). ACM, 355–364. https://doi.org/10.1145/3209978.3209981
[22] Xiangnan He, Zhankui He, Jingkuan Song, Zhenguang Liu, Yu-Gang Jiang, and Tat-Seng Chua. 2018. Nais: Neural attentive item similarity model for recommendation. *IEEE Trans. Knowl. Data Eng.* 30, 12 (2018), 2354–2366.
[23] Binbin Hu, Chuan Shi, Wayne Xin Zhao, and Philip S. Yu. 2018. Leveraging meta-path based context for top- N recommendation with A neural co-attention model. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'18)*, Yike Guo and Faisal Farooq (Eds.). ACM, 1531–1540. https://doi.org/10.1145/3219819.3219965
[24] Yan Hu, Qimin Peng, Xiaohui Hu, and Rong Yang. 2014. Time aware and data sparsity tolerant web service recommendation based on improved collaborative filtering. *IEEE Trans. Serv. Comput.* 8, 5 (2014), 782–794.

[25] Xiaowen Huang, Shengsheng Qian, Quan Fang, Jitao Sang, and Changsheng Xu. 2020. Meta-path augmented sequential recommendation with contextual co-attention network. *ACM Trans. Multim. Comput. Commun. Appl.* 16, 2 (2020), 52:1–52:24. https://doi.org/10.1145/3382180

[26] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys'10)*, Xavier Amatriain, Marc Torrens, Paul Resnick, and Markus Zanker (Eds.). ACM, 135–142.

[27] Bo Jiang. 2020. Multi-graph group collaborative filtering. In *Proceedings of the International Conference on Multimedia Retrieval (ICMR'20)*, Cathal Gurrin, Björn Þór Jónsson, Noriko Kando, Klaus Schöffmann, Yi-Ping Phoebe Chen, and Noel E. O'Connor (Eds.). ACM, 555–562. https://doi.org/10.1145/3372278.3390715

[28] Jiarui Jin, Jiarui Qin, Yuchen Fang, Kounianhua Du, Weinan Zhang, Yong Yu, Zheng Zhang, and Alexander J. Smola. 2020. An efficient neighborhood-based interaction model for recommendation on heterogeneous graph. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'20)*, Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash (Eds.). ACM, 75–84.

[29] Kyung-Min Kim, Dong-Hyun Kwak, Hanock Kwak, Young-Jin Park, Sangkwon Sim, Jae-Han Cho, Minkyu Kim, Jihun Kwon, Nako Sung, and Jung-Woo Ha. 2019. Tripartite heterogeneous graph propagation for large-scale social recommendation. In *Proceedings of the ACM RecSys 2019 Late-Breaking Results co-located with the 13th ACM Conference on Recommender Systems (RecSys'19)*, Marko Tkalcic and Sole Pera (Eds.), Vol. 2431.

[30] Irwin King, Michael R. Lyu, and Hao Ma. 2010. Introduction to social recommendation. In *Proceedings of the 19th International Conference on World Wide Web*. 1355–1356.

[31] Yehuda Koren. 2008. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Ying Li, Bing Liu, and Sunita Sarawagi (Eds.). ACM, 426–434.

[32] Yehuda Koren, Robert M. Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37. https://doi.org/10.1109/MC.2009.263

[33] Adit Krishnan, Hari Cheruvu, Tao Cheng, and Hari Sundaram. 2019. A modular adversarial approach to social recommendation. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM'19)*, Wenwu Zhu, Dacheng Tao, Xueqi Cheng, Peng Cui, Elke A. Rundensteiner, David Carmel, Qi He, and Jeffrey Xu Yu (Eds.). ACM, 1753–1762. https://doi.org/10.1145/3357384.3357898

[34] Daniel Lemire and Anna Maclachlan. 2005. Slope one predictors for online rating-based collaborative filtering. In *Proceedings of the 2005 SIAM International Conference on Data Mining (SDM'05)*, Hillol Kargupta, Jaideep Srivastava, Chandrika Kamath, and Arnold Goodman (Eds.). SIAM, 471–475.

[35] Xin Li and Hsinchun Chen. 2013. Recommendation as link prediction in bipartite graphs: A graph kernel-based machine learning approach. *Decis. Supp. Syst.* 54, 2 (2013), 880–890. https://doi.org/10.1016/j.dss.2012.09.019

[36] Tingting Liang, Liang Chen, Jian Wu, Hai Dong, and Athman Bouguettaya. 2016. Meta-path based service recommendation in heterogeneous information networks. In *Proceedings of the 14th International Conference on Service-Oriented Computing (ICSOC'16),* Lecture Notes in Computer Science, Quan Z. Sheng, Eleni Stroulia, Samir Tata, and Sami Bhiri (Eds.), Vol. 9936. Springer, 371–386. https://doi.org/10.1007/978-3-319-46295-0_23

[37] Yanxiang Ling, Weiwei Zhao, Wenjing Yang, and Fei Cai. 2018. TMP: Meta-path based recommendation on time-weighted heterogeneous information networks. In *Proceedings of the 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS'18)*. IEEE, 679–683. https://doi.org/10.1109/CCIS.2018.8691240

[38] Chun-Yi Liu, Chuan Zhou, Jia Wu, Yue Hu, and Li Guo. 2018. Social recommendation with an essential preference space. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI'18), the 30th Innovative Applications of Artificial Intelligence (IAAI'18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI'18)*, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.). AAAI Press, 346–353.

[39] Juntao Liu, Caihua Wu, and Wenyu Liu. 2013. Bayesian probabilistic matrix factorization with social relations and item contents for recommendation. *Decis. Supp. Syst.* 55, 3 (2013), 838–850. https://doi.org/10.1016/j.dss.2013.04.002

[40] Yang Liu, Chen Liang, Xiangnan He, Jiaying Peng, Zibin Zheng, and Jie Tang. 2020. Modelling high-order social relations for item recommendation. *IEEE Transactions on Knowledge and Data Engineering.* DOI: 10.1109/TKDE.2020.3039463

[41] Hao Ma, Irwin King, and Michael R. Lyu. 2009. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'09)*, James Allan, Javed A. Aslam, Mark Sanderson, ChengXiang Zhai, and Justin Zobel (Eds.). ACM, 203–210.

[42] Hao Ma, Haixuan Yang, Michael R. Lyu, and Irwin King. 2008. SoRec: Social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM'08)*, James G. Shanahan, Sihem Amer-Yahia, Ioana Manolescu, Yi Zhang, David A. Evans, Aleksander Kolcz, Key-Sun Choi, and Abdur Chowdhury (Eds.). ACM, 931–940. https://doi.org/10.1145/1458082.1458205

[43] McPherson, Miller, Smith-Lovin, Lynn, Cook, and M. James. 2001. Birds of a feather: Homophily in social networks. *Annu. Rev. Sociol.* 27, 1 (2001), 415–444.
[44] Sanjay Purushotham and Yan Liu. 2012. Collaborative topic regression with social matrix factorization for recommendation systems. In *Proceedings of the 29th International Conference on Machine Learning (ICML'12)*. Omnipress.
[45] Yanru Qu, Ting Bai, Weinan Zhang, Jianyun Nie, and Jian Tang. 2019. An end-to-end neighborhood-based interaction model for knowledge-enhanced recommendation. In *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data*. 1–9.
[46] Dimitrios Rafailidis and Fabio Crestani. 2019. Adversarial training for review-based recommendations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*, Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.). ACM, 1057–1060. https://doi.org/10.1145/3331184.3331313
[47] Ruslan Salakhutdinov and Andriy Mnih. 2007. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems 20, Proceedings of the 21st Annual Conference on Neural Information Processing Systems*, John C. Platt, Daphne Koller, Yoram Singer, and Sam T. Roweis (Eds.). Curran Associates, Inc., 1257–1264.
[48] Chuan Shi, Zhiqiang Zhang, Ping Luo, Philip S. Yu, Yading Yue, and Bin Wu. 2015. Semantic path based personalized recommendation on weighted heterogeneous information networks. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM'15)*, James Bailey, Alistair Moffat, Charu C. Aggarwal, Maarten de Rijke, Ravi Kumar, Vanessa Murdock, Timos K. Sellis, and Jeffrey Xu Yu (Eds.). ACM, 453–462. https://doi.org/10.1145/2806416.2806528
[49] Weiping Song, Zhiping Xiao, Yifan Wang, Laurent Charlin, Ming Zhang, and Jian Tang. 2019. Session-based social recommendation via dynamic graph attention networks. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining (WSDM'19)*, J. Shane Culpepper, Alistair Moffat, Paul N. Bennett, and Kristina Lerman (Eds.). ACM, 555–563. https://doi.org/10.1145/3289600.3290989
[50] Yuqi Song, Min Gao, Junliang Yu, and Qingyu Xiong. 2018. Social recommendation based on implicit friends discovering via meta-path. In *Proceedings of the IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI'18)*, Lefteri H. Tsoukalas, Éric Grégoire, and Miltiadis Alamaniotis (Eds.). IEEE, 197–204. https://doi.org/10.1109/ICTAI.2018.00039
[51] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S. Yu, and Tianyi Wu. 2011. PathSim: Meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endow.* 4, 11 (2011), 992–1003. http://www.vldb.org/pvldb/vol4/p992-sun.pdf.
[52] Jiliang Tang, Xia Hu, and Huan Liu. 2013. Social recommendation: A review. *Soc. Netw. Anal. Min.* 3, 4 (2013), 1113–1133.
[53] M Vijaikumar, Shirish Shevade, and M Narasimha Murty. 2019. SoRecGAT: Leveraging graph attention mechanism for top-n social recommendation. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 430–446.
[54] Hongwei Wang, Jia Wang, Jialin Wang, Miao Zhao, Weinan Zhang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2017. Graphgan: Graph representation learning with generative adversarial nets. arXiv:1711.08267. Retrieved from https://arxiv.org/abs/1711.08267.
[55] Meng Wang, Kuiyuan Yang, Xian-Sheng Hua, and Hong-Jiang Zhang. 2010. Towards a relevant and diverse search of social images. *IEEE Trans. Multimedia* 12, 8 (2010), 829–842.
[56] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'19)*, Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis (Eds.). ACM, 950–958.
[57] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*, Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.). ACM, 165–174.
[58] Xiaodong Wang, Zhen Liu, Nana Wang, and Wentao Fan. 2020. Relational metric learning with dual graph attention networks for social recommendation. In *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 104–117.
[59] Yang Wang. 2021. Survey on deep multi-modal data analytics: Collaboration, rivalry, and fusion. *ACM Trans. Multimedia Comput. Commun. Appl.* 17, 1s (2021), 1–25.
[60] Yang Wang, Xuemin Lin, Lin Wu, and Wenjie Zhang. 2017. Effective multi-query expansions: Collaborative deep networks for robust landmark retrieval. *IEEE Trans. Image Process.* 26, 3 (2017), 1393–1404. https://doi.org/10.1109/TIP.2017.2655449

[61] Jiahui Wen, Jingwei Ma, Hongkui Tu, Mingyang Zhong, Guangda Zhang, Wei Yin, and Jian Fang. 2020. Hierarchical text interaction for rating prediction. *Knowl. Based Syst.* 206 (2020), 106344.

[62] Qiong Wu, Yong Liu, Chunyan Miao, Binqiang Zhao, Yin Zhao, and Lu Guan. 2019. PD-GAN: Adversarial learning for personalized diversity-promoting recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19)*, Sarit Kraus (Ed.). 3870–3876. https://doi.org/10.24963/ijcai.2019/537

[63] Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Peng He, Paul Weng, Han Gao, and Guihai Chen. 2019. Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. In *Proceedings of the World Wide Web Conference (WWW'19)*, Ling Liu, Ryen W. White, Amin Mantrach, Fabrizio Silvestri, Julian J. McAuley, Ricardo Baeza-Yates, and Leila Zia (Eds.). ACM, 2091–2102. https://doi.org/10.1145/3308558.3313442

[64] Yang Xiao, Lina Yao, Qingqi Pei, Xianzhi Wang, Jian Yang, and Quan Z. Sheng. 2020. MGNN: Mutualistic graph neural network for joint friend and item recommendation. *IEEE Intell. Syst.* 35, 5 (2020), 7–17.

[65] Hongzhi Yin, Bin Cui, Ling Chen, Zhiting Hu, and Chengqi Zhang. 2015. Modeling location-based user rating profiles for personalized recommendation. *ACM Trans. Knowl. Discov. Data* 9, 3 (2015), 19:1–19:41. https://doi.org/10.1145/2663356

[66] Junliang Yu, Min Gao, Hongzhi Yin, Jundong Li, Chongming Gao, and Qinyong Wang. 2019. Generating reliable friends via adversarial training to improve social recommendation. In *2019 Proceedings of the IEEE International Conference on Data Mining (ICDM'19)*, Jianyong Wang, Kyuseok Shim, and Xindong Wu (Eds.). IEEE, 768–777. https://doi.org/10.1109/ICDM.2019.00087

[67] Junliang Yu, Hongzhi Yin, Jundong Li, Min Gao, Zi Huang, and Lizhen Cui. 2020. Enhance social recommendation with adversarial graph convolutional networks. arXiv:2004.02340. Retrieved from https://arxiv.org/abs/2004.02340.

[68] Chengyuan Zhang, Jiayu Song, Xiaofeng Zhu, Lei Zhu, and Shichao Zhang. 2021. HCMSL: Hybrid cross-modal similarity learning for cross-modal retrieval. *ACM Trans. Multimedia Comput. Commun. Appl.* 17, 1s (2021), 1–22.

[69] Jiani Zhang, Xingjian Shi, Shenglin Zhao, and Irwin King. 2019. STAR-GCN: Stacked and reconstructed graph convolutional networks for recommender systems. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19)*, Sarit Kraus (Ed.). 4264–4270. https://doi.org/10.24963/ijcai.2019/592

[70] Huan Zhao, Quanming Yao, Jianda Li, Yangqiu Song, and Dik Lun Lee. 2017. Meta-graph based recommendation fusion over heterogeneous information networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 635–644. https://doi.org/10.1145/3097983.3098063

[71] YiMing Zheng, Kexin Zhao, and Antonis C. Stylianou. 2011. The formation of social influence in online recommendation systems: A study of user reviews on amazon.com. In *Proceedings of the International Conference on Information Systems (ICIS'11)*, Dennis F. Galletta and Ting-Peng Liang (Eds.). Association for Information Systems.

[72] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2018. Graph neural networks: A review of methods and applications. arXiv:1812.08434. Retrieved from https://arxiv.org/abs/1812.08434.

[73] Yu Zhu, Yu Gong, Qingwen Liu, Yingcai Ma, Wenwu Ou, Junxiong Zhu, Beidou Wang, Ziyu Guan, and Deng Cai. 2019. Query-based interactive recommendation by meta-path and adapted attention-GRU. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM'19)*, Wenwu Zhu, Dacheng Tao, Xueqi Cheng, Peng Cui, Elke A. Rundensteiner, David Carmel, Qi He, and Jeffrey Xu Yu (Eds.). ACM, New York, NY, 2585–2593. https://doi.org/10.1145/3357384.3357805

Received October 2020; revised May 2021; accepted May 2021