A Survey of Graph Neural Networks for Social Recommender Systems

KARTIK SHARMA*, Georgia Institute of Technology, USA
YEON-CHANG LEE*, Georgia Institute of Technology, USA
SIVAGAMI NAMBI, Georgia Institute of Technology, USA
ADITYA SALIAN†, Georgia Institute of Technology, USA
SHLOK SHAH†, Georgia Institute of Technology, USA
SANG-WOOK KIM, Hanyang University, South Korea
SRIJAN KUMAR+, Georgia Institute of Technology, USA

Social recommender systems (SocialRS) simultaneously leverage the user-to-item interactions as well as the user-to-user social relations for the task of generating item recommendations to users. Additionally exploiting social relations is clearly effective in understanding users' tastes due to the effects of homophily and social influence. For this reason, SocialRS has increasingly attracted attention. In particular, with the advance of graph neural networks (GNN), many GNN-based SocialRS methods have been developed recently. Therefore, we conduct a comprehensive and systematic review of the literature on GNN-based SocialRS.

In this survey, we first identify 80 papers on GNN-based SocialRS after annotating 2,151 papers by following the PRISMA framework (preferred reporting items for systematic reviews and meta-analyses). Then, we comprehensively review them in terms of their inputs and architectures to propose a novel taxonomy: (1) input taxonomy includes 5 groups of input type notations and 7 groups of input representation notations; (2) architecture taxonomy includes 8 groups of GNN encoder notations, 2 groups of decoder notations, and 12 groups of loss function notations. We classify the GNN-based SocialRS methods into several categories as per the taxonomy and describe their details. Furthermore, we summarize benchmark datasets and metrics widely used to evaluate the GNN-based SocialRS methods. Finally, we conclude this survey by presenting some future research directions.

CCS Concepts: • **Computing methodologies → Neural networks**; • **Information systems → Social networks**; **Recommender systems**.

Additional Key Words and Phrases: graph neural networks, social network, recommender systems, social recommendation, survey

---

* Equal Contribution.
† Equal Contribution.
+ Corresponding author.

Authors' addresses: Kartik Sharma*, ksartik@gatech.edu, Georgia Institute of Technology, USA; Yeon-Chang Lee*, yeonchang@gatech.edu, Georgia Institute of Technology, USA; Sivagami Nambi, sivagami.nambi@gatech.edu, Georgia Institute of Technology, USA; Aditya Salian†, asalian@gatech.edu, Georgia Institute of Technology, USA; Shlok Shah†, sshah672@gatech.edu, Georgia Institute of Technology, USA; Sang-Wook Kim, wook@hanyang.ac.kr, Hanyang University, South Korea; Srijan Kumar+, srijan@gatech.edu, Georgia Institute of Technology, USA.

XXXX-XXXX/2022/12-ART $15.00
https://doi.org/XXXXXXX.XXXXXXX

ACM Reference Format:
Kartik Sharma*, Yeon-Chang Lee*, Sivagami Nambi, Aditya Salian†, Shlok Shah†, Sang-Wook Kim, and Srijan Kumar+. 2022. A Survey of Graph Neural Networks for Social Recommender Systems. 1, 1 (December 2022), 31 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

With the advent of online social network platforms (*e.g.*, Facebook, Twitter, Instagram, etc.), there has been a surge of research efforts in developing social recommender systems (SocialRS), which simultaneously utilize user-user social relations along with user-item interactions to recommend relevant items to users. Exploiting social relations in recommendation works well because of the effects of *social homophily* [61] and *social influence* [60]: (1) social homophily indicates that a user tends to connect herself to other users with similar attributes and preferences, and (2) social influence indicates that users with direct or indirect relations tend to influence each other to make themselves become more similar. Accordingly, SocialRS can effectively mitigate the data sparsity problem by exploiting social neighbors to capture the preferences of a sparsely interacting user.

Literature has shown that SocialRS can be applied successfully in various recommendation domains (*e.g.*, product [101, 103], music [116–118], location [39, 72, 100], and image [86, 99, 102]), thereby improving user satisfaction. Furthermore, techniques and insights explored from SocialRS can also be exploited in real-world applications other than recommendations. For instance, García-Sánchez et al. [20] leveraged SocialRS to design a decision-making system for marketing (*e.g.*, advertisement), while Gasparetti et al. [21] analyzed SocialRS in terms of community detection.

Motivated by such wide applicability, there has been an increasing interest in research on developing accurate SocialRS models. In the early days, research focused on matrix factorization (MF) techniques [28, 54–57, 84, 112]. However, MF-based methods cannot effectively model the complex (*i.e.*, non-linear) relationships inherent in user-user social relations and user-item interactions [76]. Motivated by this, most recent works have focused on applying deep-learning techniques to SocialRS, *e.g.*, autoencoder [11, 115], generative adversarial networks (GAN) [35], and graph neural networks (GNN) [16, 102].

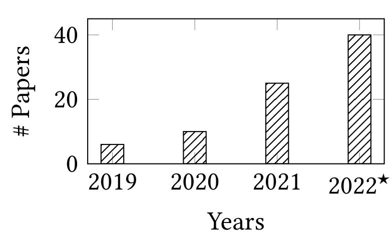

Fig. 1. The number of papers related to GNN-based SocialRS per year. ★ For 2022, we count the number of relevant papers published until October.

In particular, since user-item interactions and user-user social relations can naturally be represented as graph data, GNN-based SocialRS has increasingly attracted attention in the literature. As a demonstration, Figure 1 shows that the number of papers related to GNN-based SocialRS has increased consistently since 2019. Given the growing and timely interest in this area, we survey GNN-based SocialRS methods in this survey.

1.1 Challenges

Applying GNN into SocialRS is not trivial and faces the following challenges.

Input representation. The input data should be modeled appropriately into a heterogeneous graph structure. Many SocialRS methods build two separate graphs: one where nodes represent users and items, and edges represent user-item interactions; the other where nodes represent users and edges represent user-user social relations. Thus, GNN methods for SocialRS need to extract knowledge from both the networks simultaneously for accurate inference. This is in contrast with most regular GNNs that consider only a single network. Additionally, we note that there are valuable input features in the two networks, such as user/item attributes, item knowledge/relation,

Table 1. Comparison with existing surveys. For each survey, we summarize the topics covered, some statistics regarding GNN-based SocialRS papers (*i.e.*, relevant papers), and the main scope to survey.

Surveys	Topics		GNN-based SocialRS Papers		Scope
	SocialRS	GNN	# Papers	Latest Year	
[7, 14, 85, 109, 114]	✓		0	-	Traditional SocialRS
[21]	✓		0	-	SocialRS for CD
[77]	✓		0	-	General SocialRS
[76]	✓	✓ (partial)	1	2019	General SocialRS
[12]	✓ (partial)	✓ (partial)	2	2019	Graph-based RS
[94]	✓ (partial)	✓ (partial)	3	2020	Graph-based RS
[104]	✓ (partial)	✓	14	2021	GNN-based RS
[19]	✓ (partial)	✓	19	2021	GNN-based RS
Ours	✓	✓	80	Oct, 2022	GNN-based SocialRS

✓: fully covered, ✓ (partial): partially covered.
Papers: the number of GNN-based SocialRS papers included in the survey.
Latest year: the latest publication year of a relevant paper included in the survey.

and group information. Thus, methods fuse features along with network information in GNN-based SocialRS. In this survey, we discuss the input types used in GNN-based SocialRS methods and the different ways they are represented as graphs.

Design of GNN encoder. The performance of GNN-based SocialRS methods relies heavily on their GNN encoders, which aim to represent users and items into low-dimensional embeddings. For this reason, existing SocialRS methods have explored various design choices regarding GNN encoders and have adopted different architectures according to their goals. For instance, many SocialRS methods employ the graph attention neural network (GANN) [88] to differentiate each user's preference for items or each user's influence on their social friends. On the other hand, some methods [22, 65, 66, 82, 111] use the graph recurrent neural networks (GRNN) [68, 120] to model the sequential behaviors of users. It should be noted that GNN encoders for SocialRS need to simultaneously consider the characteristics of user-item interactions and user-user social relations. This is in contrast with GNN encoders for non-SocialRS that model only user-item interactions. In this survey, we discuss different types of GNN encoders used by SocialRS methods.

Training. The training of GNN-based SocialRS should be designed to reflect users' tastes and items' characteristics in the embeddings for the corresponding users and items. To this end, SocialRS methods employ well-known loss functions, such as mean squared error (MSE), Bayesian personalized ranking (BPR) [70], and cross-entropy (CE), to reconstruct user behaviors. Furthermore, to mitigate the data sparsity problem, some works have additionally employed auxiliary loss functions such as self-supervised loss [49] and group-based loss [36, 42]. It is worth mentioning that loss functions used by GNN-based SocialRS are designed so that rich structural information such as motifs and user attributes can be exploited. These are not considered by loss functions for non-SocialRS. In this survey, we discuss the training remedies of GNN-based SocialRS methods to learn the user and item embeddings.

1.2 Related Surveys

Most of the existing surveys, which fully cover SocialRS papers, focus either on traditional methods [7, 14, 67, 75, 85, 109, 114] (*e.g.*, matrix factorization), feature information [77] (*e.g.*, context),

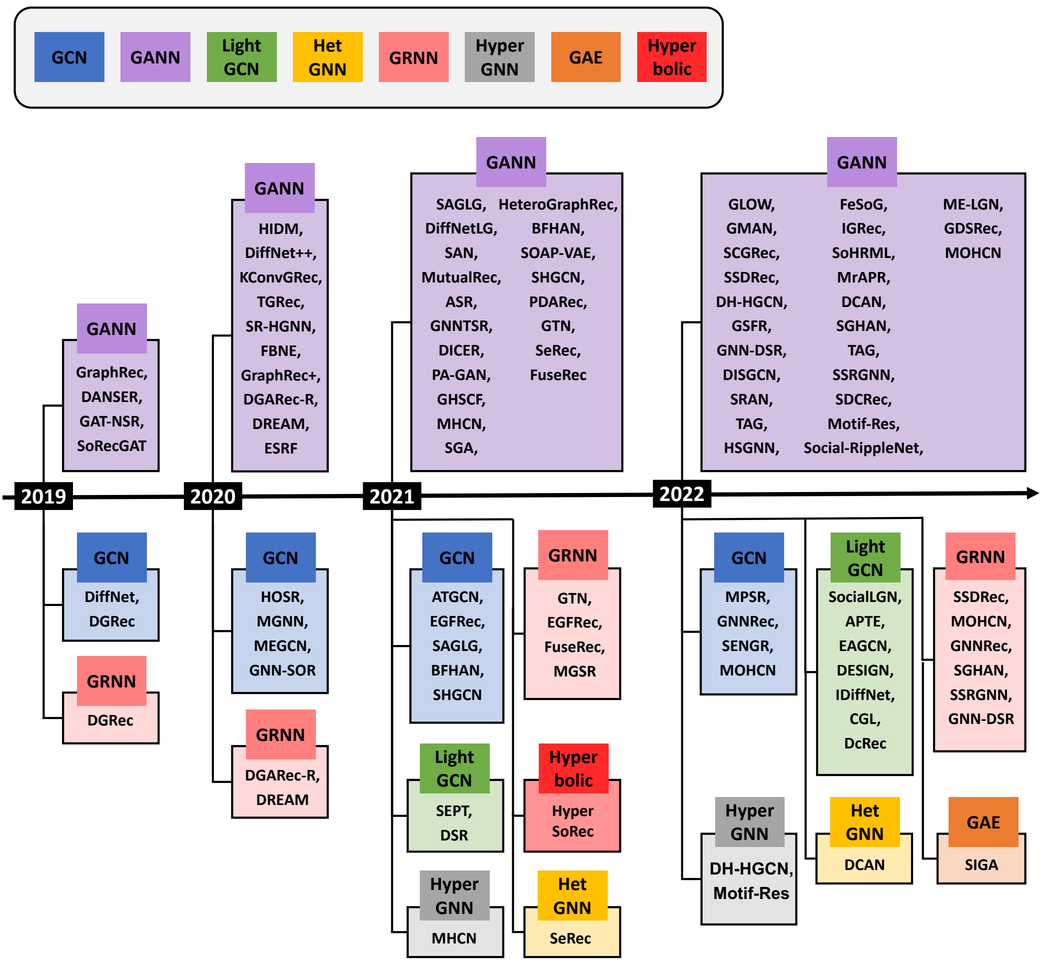

Fig. 2. A timeline of GNN-based SocialRS methods. We categorize methods according to their GNN encoders: graph convolutional network (GCN), lightweight GCN (LightGCN), graph attention neural networks (GANN), heterogeneous GNN (HetGNN), graph recurrent neural networks (GRNN), hypergraph neural networks (HyperGNN), graph autoencoder (GAE), and hyperbolic GNN. It should be noted that some methods employ two or more GNN encoders in their architectures.

or a specific application [21] (*e.g.*, community detection). On the other hand, the other related surveys [12, 19, 94, 104] focus on graph-based recommender systems, including GNN-based RS methods, but they partially cover SocialRS papers in their surveys. A comparison between the current survey and the previous surveys is shown in Table 1.

Specifically, several survey papers on SocialRS have been published before 2019 [7, 14, 67, 75, 85, 109, 114]. However, they only focus on traditional methods such as matrix factorization and collaborative filtering. These surveys largely ignore methods that use modern-day deep-learning techniques, in particular GNN.

More recent surveys discuss the taxonomy of social recommendation, starting the comparison of deep-learning based techniques [21, 76, 77]. However, Shokeen and Rana [77] only focus on the taxonomy of feature information regarding social relations, such as context, trust, and group, used in SocialRS methods, while Gasparetti et al. [21] only discuss SocialRS methods using community

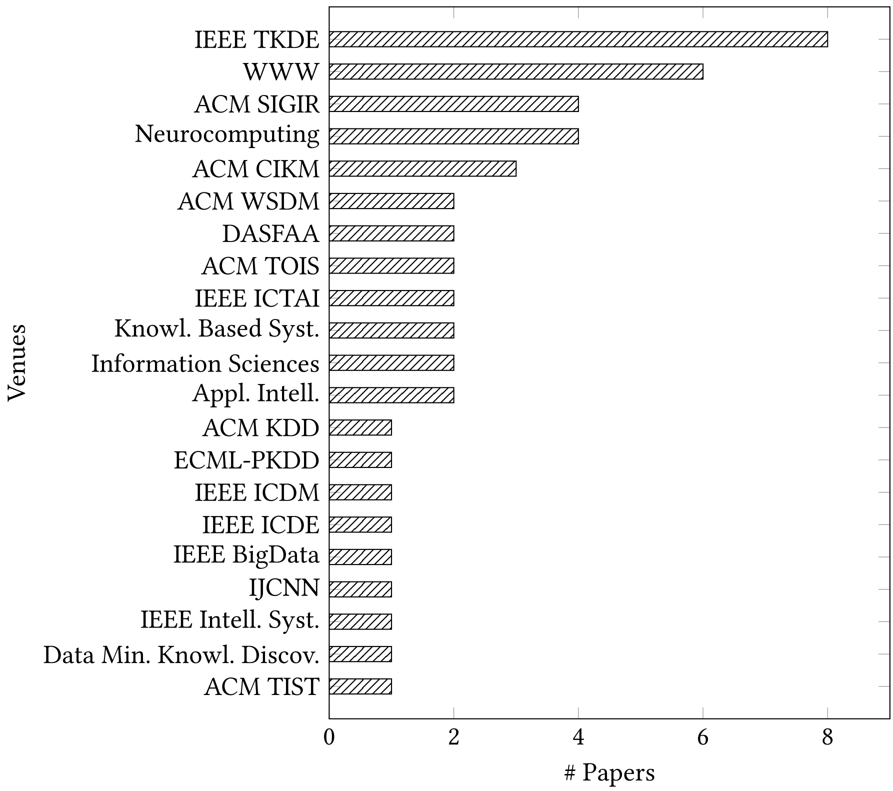

Fig. 3. The number of GNN-based SocialRS papers published in relevant journals and conferences. We only present statistics with respect to prominent data mining journals (including IEEE TKDE, ACM TOIS, Knowledge-Based Systems, and Information Sciences) and conferences (including WWW, ACM SIGIR, ACM KDD, ACM CIKM, ACM WSDM, IEEE ICDE, and IEEE ICDM). We believe it would help researchers in this field to identify appropriate venues where GNN-based SocialRS papers are published.

detection (CD) techniques. Shokeen and Rana [76] include just one social recommendation method based on GNNs.

With the advent of GNNs in recommender systems, multiple surveys have been conducted on graph-based recommender systems [12, 19, 94, 104]. However, their focus is not on SocialRS as they consider different kinds of recommender systems where graph-learning is employed. They cover only a small section of most representative papers on GNN-based SocialRS. Thus, one cannot rely on these surveys to gain insights on the ever-increasing field of using GNNs for SocialRS.

As shown in Table 1, no survey paper exists in the literature that focuses specifically on GNN-based SocialRS methods. In the current work, we aim to fill this gap by providing a comprehensive and systematic survey on GNN-based SocialRS methods.

1.3 Contributions

The main contribution of this survey paper is summarized as follows:

- **The First Survey in GNN-based SocialRS**: To the best of our knowledge, we are the first to systematically dedicate ourselves to reviewing GNN-based SocialRS methods. Most of the existing surveys focus either on traditional methods [7, 14, 67, 75, 85, 109, 114] (*e.g.*, matrix factorization), feature information [77] (*e.g.*, context), or a specific application [21] (*e.g.*, community detection). The other related surveys [12, 19, 94, 104] focus on graph-based recommender systems, but they partially cover SocialRS.

- **Comprehensive Survey**: We systematically identify the relevant papers on GNN-based SocialRS by following the guidelines of the preferred reporting items for systematic reviews and meta-analyses (PRISMA framework) [63]. Then, we comprehensively review them in terms of their inputs and architectures. Figure 2 provides a brief timeline of GNN-based SocialRS methods. In addition, Figure 3 shows the number of relevant papers published in relevant journals (*e.g.*, IEEE TKDE and ACM TOIS) and conferences (*e.g.*, WWW, ACM SIGIR, and ACM CIKM).
- **Novel Taxonomy of Inputs and Architectures**: We provide a novel taxonomy of inputs and architectures in GNN-based SocialRS methods, enabling researchers to capture the research trends in this field easily. An input taxonomy includes 5 groups of input type notations and 7 groups of input representation notations. On the other hand, an architecture taxonomy includes 8 groups of GNN encoder notations, 2 groups of decoder notations, and 12 groups (4 for primary losses and 8 for auxiliary losses) of loss function notations.
- **Benchmark Datasets**: We review 17 benchmark datasets used to evaluate the performance of GNN-based SocialRS methods. We group the datasets into 8 domains (*i.e.*, product, location, movie, image, music, bookmark, microblog, and miscellaneous). Also, we present some statistics for each dataset and a list of papers using the dataset.
- **Future Directions**: We discuss the limitations of existing GNN-based SocialRS methods and provide several future research directions.

The rest of this survey paper is organized as follows. In Section 2, we introduce the survey methodology based on PRISMA [63] that collects the papers on GNN-based SocialRS thoroughly. In Section 3, we define the social recommendation problem. In Sections 4 and 5, we review 80 GNN-based SocialRS methods in terms of their inputs and architectures, respectively. We summarize 17 benchmark datasets and 8 evaluation metrics, widely-used in GNN-based SocialRS methods, in Section 6. Section 7 discusses future research directions. Finally, we conclude the paper in Section 8.

2 SURVEY METHODOLOGY

Following the guidelines set by the PRISMA [63], the Scopus index was queried to filter for relevant literature. In particular, the following query was run on October 14, 2022, resulting in 2,151 papers.

```
TITLE-ABS-KEY (social AND (recommendation OR recommender) AND graph) AND (PUBYEAR > 2009) AND
(LIMIT-TO (LANGUAGE, “English”))
```

To obtain the final list of relevant papers for the current survey, an iterative strategy of manual reviewing and filtering was carried out, following PRISMA guidelines. Four expert annotators were used to select the relevant papers. Before reviewing the papers, a comprehensive and exhaustive discussion was held among the annotators to discuss and agree upon the definitions of the main concepts that a paper is to be examined for before including it in the survey. These included concepts of Graph Neural Networks and Social Recommendation.

Based on these guidelines, each annotator labeled one batch of 200 papers together. Each paper in this batch was assigned one of the three categories by each annotator: “Yes”, “No”, and “Maybe”. “Yes” represents full confidence of relevance, “Maybe” represents some confidence of relevance, and “No” represents full confidence of irrelevance of the paper for the current survey. A high inter-annotator agreement of 0.845 among the annotators was reported on this set.

The remaining papers were then divided equally among the annotators without any overlap. The annotator assigned each paper a label of “Yes”, “No”, or “Maybe”. Papers marked “Maybe” were

Table 2. Notations used in this paper.

Notation	Description
$\mathcal{U}, \mathcal{I}$	Sets of users p_i and items q_j
$\mathbf{R}$, $\mathbf{S}$	Matrices representing U-I rating and U-U social
$\mathcal{N}_{p_i}$	Set of items rated by p_i
$\mathbf{u}_i^I$	Embedding of p_i obtained via the user interaction encoder
$\mathbf{u}_i^S$	Embedding of p_i obtained via the user social encoder
$\mathbf{u}_i$	Embedding of p_i obtained by fusing $\mathbf{u}_i^I$ and $\mathbf{u}_i^S$
$\mathbf{v}_j$	Embedding of q_j via the item encoder
r_{ij}	Real rating score of p_i on q_j
$\hat{r}_{ij}$	Predicted preference of p_i on q_j via the decoder

reviewed again by the other annotators to reach a consensus. Finally, papers marked "Yes" were collected together and these served as the focus of our survey. Through this comprehensive process of filtering, we finally found 80 papers that study GNN-based SocialRS for our survey paper.

3 NOTATIONS AND PROBLEM DEFINITION

The social recommendation problem is formulated as follows. Let $\mathcal{U} = \{p_1, p_2, \cdots, p_m\}$ and $\mathcal{I} = \{q_1, q_2, \cdots, q_n\}$ be sets of m users and n items, respectively. Also, $\mathbf{R} \in \mathbb{R}^{m \times n}$ represents a rating matrix that stores user-item ratings (that we call U-I rating). $\mathbf{S} \in \mathbb{R}^{m \times m}$ represents a social matrix that stores user-user social relations (that we call U-U social). In addition, $\mathcal{N}_{p_i}$ indicates a set of items rated by a user p_i. In this paper, we use bold uppercase letters and bold lowercase letters to denote matrices and vectors, respectively. Also, we use calligraphic letters to denote sets and graphs. Table 2 summarizes a list of notations used in this paper.

The goal of GNN-based SocialRS methods is to solve the rating prediction and/or top-N recommendation tasks. Given $\mathbf{R}$ and $\mathbf{S}$, both tasks are formally defined as follows:

Problem 1 (Rating Prediction). *The goal is to predict the rating values for unrated items (i.e., $\mathcal{I} \setminus \mathcal{N}_{p_i}$) in $\mathbf{R}$ as close as possible to the ground truth.*

Problem 2 (Top-N Recommendation). *The goal is to recommend the top-N items that are most likely to be preferred by each user p_i among p_i's unrated items (i.e., $\mathcal{I} \setminus \mathcal{N}_{p_i}$).*

4 TAXONOMY OF INPUTS

In this section, we present a taxonomy of inputs for GNN-based SocialRS. Figures 4 and 5 depict the input types and their representations, respectively. In the subsequent subsections, we describe each of these in detail.

4.1 Input Types: Types of Inputs to the Models

In this subsection, we group the input types used by GNN-based SocialRS into 5 categories: user-item ratings, user-user social relations, attributes, knowledge graph (KG), and groups. Table 3 categorizes all papers based on the input data types they use.

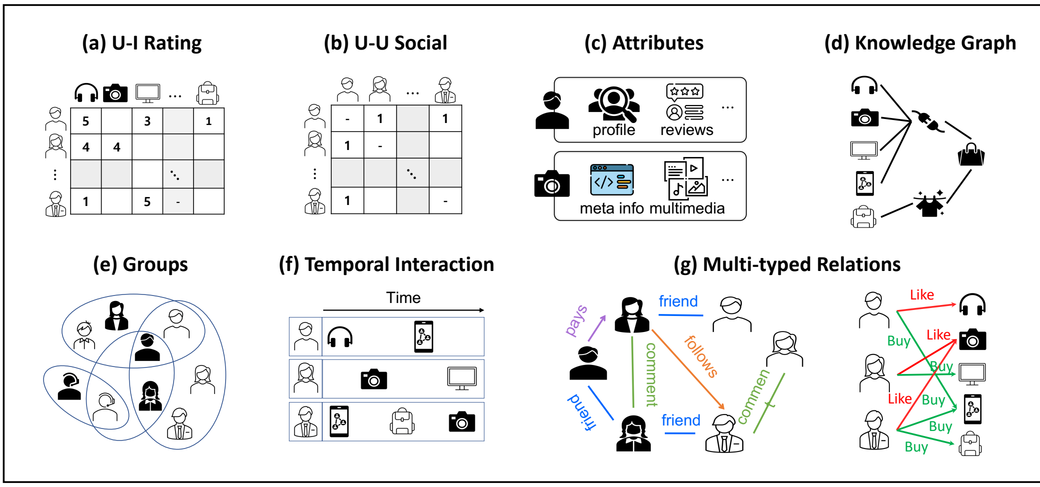

Fig. 4. Overview of input types used by GNN-based SocialRS methods.

4.1.1 *User-Item Rating.*

Users interact with different items as they rate them, thus forming the rating matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$. Therefore, each user has a list of items that he/she has interacted with along with the corresponding rating.

The timestamp of the user-item interaction may also be available and can be exploited to recommend items to users at specific points in time. Each rating can thus also be associated with a timestamp for that rating. Some models exploit the temporal information to make more-effective recommendations in continuous time [1, 65, 82] or during a user session [22, 80, 81, 111].

Furthermore, one may also have multi-typed user-item interactions. For example, a user may interact with an item positively (positive rating) or negatively (negative rating). Some models have distinguished among these different interaction types to predict each type more effectively [110].

4.1.2 *User-User Social.*

The second essential input to SocialRS is the social adjacency matrix $\mathrm{S} \in \mathbb{R}^{m \times m}$, storing user-user social relations.

People may be connected to each other via different kinds of social relations. For example, two users may be related if they are friends or if they may co-comment on an item or if one follows the other, etc. DH-HGCN [24] and BFHAN [124] consider multifaceted, heterogeneous user-user relations in the social network.

4.1.3 *Additional Features.*

Attributes. Both user and items may have additional attributes that can be encoded by the models to make better social recommendations. User attributes are often features of user profiles on social media, *e.g.*, age, sex, etc., while item attributes are often information about the items such as its price and category. Some models just incorporate user attributes [72, 106], some only item attributes [8, 23], and others incorporate both [30, 33, 78, 101, 102].

Knowledge Graph (KG). Items are structured on a product site in the form of a knowledge graph where items are related with each other if they have some mutual dependency. Models incorporate such dependencies between items as represented by this knowledge graph [71, 87].

Groups. Users are often grouped together denoting a group structure among them. For instance, multiple users can form an online social group based on similar interests or hobbies. Models incorporate the group membership in addition to the social relations to model the social network

Table 3. Taxonomy of input types.

U-I Rating	U-U Social	Additional Features		Models
		User	Item	
Static	Homogeneous	-	-	GraphRec [16], DANSER [103], DICER [18], ASR [32], GNNTSR [58], GAT-NSR [64], SoRecGAT [89], SAGLG [48], PA-GAN [27], MGNN [107], MutualRec [106], GHSCF [2], HIDM [40], SocialLGN [43], SOAP-VAE [90], GraphRec+ [17], DSR [73], SHGCN [130], GTN [26], PDARec [127], MHCN [118], SEPT [116], DcRec [100], GSFR [105], APTE [126], EAGCN [99], HOSR [50], SDCRec [15], SoHRML [51], DESIGN [86], HyperSoRec [91], CGL [123], DISGCN [38], ME-LGN [62], GDSRec [6], SGA [53], SIGA [47], ESRF [117], Motif-Res [83], FeSoG [52]
			KG	KConvGraphRec [87], HeteroGraphRec [71], Social-RippleNet [31], SCGRec [113]
			Attributes	GNN-SOR [23], MPSR [46], FBNE [5]
		Attributes	Attributes	Diffnet [102], Diffnet++ [101], DiffnetLG [78], IDiffNet [41], MEGCN [33], SAN[30], SRAN [108], MrAPR [79], SENGR [74], TAG [69], ATGCN [72], HSGNN [97]
		Groups	-	IGRec [10], GLOW [36]
			Attributes	GMAN [42]
	Multiple	-	Attributes	DH-HGCN [24]
		Attributes	Attributes	BFHAN [124]
Temporal	Homogeneous	-	-	EGFRec [22], FuseRec [65], DGARec-R [82], MGSR [66], MOHCN [96], GNNRec [45], DCAN [92], SGHAN [98], SSRGNN [9], GNN-DSR [44], DGRec [81], DREAM [80], SeRec [8], TGRec [1]
			KG	SSDRec[111]
Multiple	Homogeneous	-	-	SR-HGNN[110]

more effectively [10, 36, 42]. User groups can also be formed based on the businesses that they are part of or are clients of, as in [5].

4.2 Input Representations: Representation of Inputs within the Models

In order to effectively use the available inputs with GNN-based models, SocialRS methods represent them as different graphs. In particular, the input representations employed by GNN-based SocialRS can be grouped into 7 categories: U-U/U-I graphs, U-U-I graph, attributed graph, multiplex graph, U-U/U-I/I-I graphs, hypergraph, and decentralized. Table 4 categorizes papers based on the input representation they develop using the input data.

4.2.1 ***U-U/U-I Graphs.*** The simplest representation of the input for social recommendation is to use separate graphs for a user-user social network and a user-item interaction network. The user-item interaction network is represented as a bipartite graph and the user-user social network is represented as a general undirected/directed graph. Information from the two graphs is encoded

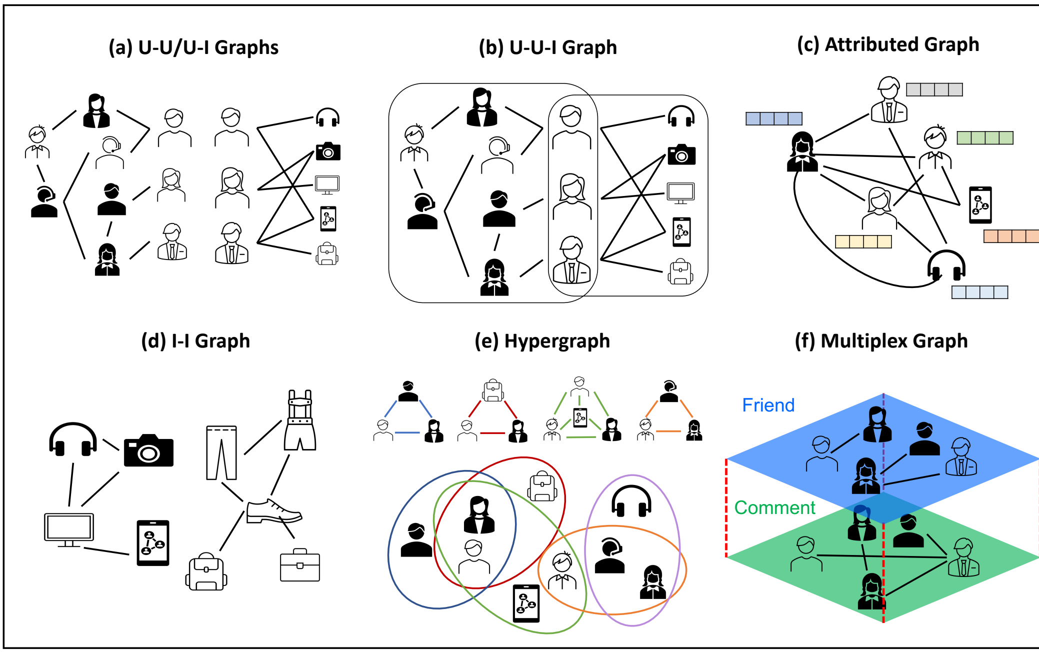

Fig. 5. Overview of input representations used by GNN-based SocialRS methods.

separately at the common user node and later aggregated. Most works follow this representation to encode users and items [16–18, 40, 78, 86, 99, 101–103, 106, 107].

4.2.2 ***U-U-I Graph***. Both kinds of user-user relations and user-item interactions can be modeled together by a single graph as well. Here, user-user edges and user-item edges in the graph need to be distinguished by the type of the end node. Many works thus merge the social relation edges and interaction edges together in a single graph to obtain node embeddings for both users and items [9, 62, 74, 130].

4.2.3 ***Attributed Graph***. Both user and item nodes may further contain features describing the corresponding entity. For example, users may have profile features while items may have their description features. These features are first encoded numerically and then represented explicitly as node attributes in the U-U/U-I graph or U-U-I graph to make effective recommendations [30, 72, 78, 79, 101, 102, 108]. These attributes are either fused with the learned embeddings or are used as initialization for the GNN layers.

4.2.4 ***Multiplex Graph***. Users may be related to each other via multiple relationships while they may also interact with items in multiple ways. Such relationships are often represented using a multiplex network, *i.e.*, using multiple layers of the U-U/U-I graph, where each layer represents a particular relation type [24, 110, 124].

4.2.5 ***U-U/U-I/I-I Graphs***. When information on item-item relations is available, an item-item knowledge graph is considered in addition to the U-U and U-I graphs. Item embeddings are now obtained separately one from the U-I interaction graph and the other from the item-item knowledge graph and then aggregated later to obtain the final item embedding [71, 87].

Table 4. Taxonomy of input representations.

Graph Representations	Models
U-U/U-I	GraphRec [16], DANSER [103], DICER [18], ASR [32], GNNTSR [58], GAT-NSR [64], SoRecGAT [89], SAGLG [48], PA-GAN [27], MGNN [107], MutualRec [106], GHSCF [2], HIDM [40], SocialLGN [43], SOAP-VAE [90], GraphRec+ [17], DSR [73], GTN [26], PDARec [127], SEPT [116], DcRec [100], GSFR [105], APTE [126], EAGCN [99], HOSR [50], SDCRec [15], SoHRML [51], DESIGN [86], HyperSoRec [91], CGL [123], DISGCN [38], GDSRec [6], SGA [53], SIGA [47], ESRF [117], SR-HGNN [110], EGFRec [22], FuseRec [65], DGARec-R [82], MGSR [66], GNNRec [45], DCAN [92], SGHAN [98], GNN-DSR [44], DGRec [81], DREAM [80], SeRec [8], TGRec [1]
U-U-I	SHGCN [130], SSRGNN [9], ME-LGN [62], SENGR [74], IGRec [10], GLOW [36],
Attributed	DiffNet [102], DiffNet++ [101], DiffNet-LG [78], IDiffNet [41], MEGCN [33], SAN [30], SRAN [108], MrAPR [79], SENGR [74], TAG [69], ATGCN [72], HSGNN [97], GMAN [42], GNN-SOR [23], MPSR [46], FBNE [5], DH-HGCN [24], BFHAN [124]
Multiplex	DH-HGCN [24], BFHAN [124]
U-U/U-I/I-I	KConvGraph [87], HeteroGraphRec [71], Social-RippleNet [31], SCGRec [113], SSDRec [111]
Hypergraph	DH-HGCN [24], MHCN [118], SHGCN [130], Motif-Res [83], MOHCN [96]
Decentralized	FeSoG [52]

4.2.6 ***Hypergraph***. One may want to incorporate higher-order relations among users and items to explicitly establish organizational properties in the input such as (1) constructing a user-only hyperedge if a group of users are connected together in closed motifs, (2) constructing a user-item joint hyperedge if a group of users interacts with the same item, and (3) constructing an item-item hyperedge if one user interacts with a group of items. Models have been developed to include just user-item joint hyperedges [130], both user-user and user-item joint hyperedges [118], and user-user and item-item hyperedges [24].

4.2.7 ***Decentralized***. Centralized data storage is becoming infeasible in practice due to rising privacy concerns. Thus, instead of storing the complete U-U/U-I graphs together, a decentralized storage of the graphs is often required. Here, the edges for social relations and interactions of each user are stored locally at each user's local server such that only non-sensitive data is shared with the centralized server [52]

5 TAXONOMY OF ARCHITECTURES

In this section, we present the taxonomy of architectures for GNN-based SocialRS. Model architectures consist of three key components as shown in Figure 6: (C1) encoders; (C2) decoders; (C3) loss functions. In (C1), the encoders represent users and items into the low-dimensional vectors

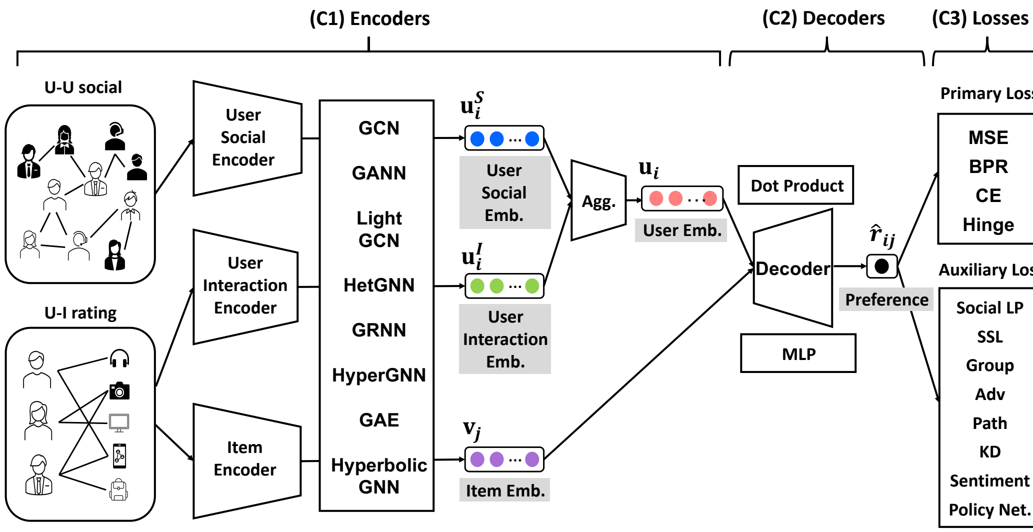

Fig. 6. Overview of architectures for GNN-based SocialRS methods.

(*i.e.*, embeddings) by employing different GNN encoders. Here, some works exploit additional information of users and/or items (*e.g.*, their attributes and groups; refer to Section 4) to construct more-accurate user and item embeddings. In (C2), the decoders predict each user's preference on each item via different operations on the user and item embeddings obtained from (C1). Finally, in (C3), different loss functions are optimized to learn the embeddings in an end-to-end manner. In the subsequent subsections, we describe each component of GNN-based SocialRS in detail.

5.1 Encoders

We group the encoders of GNN-based SocialRS into 8 categories: graph convolutional network (GCN), lightweight GCN (LightGCN), graph attention neural networks (GANN), heterogeneous GNN (HetGNN), graph recurrent neural networks (GRNN), hypergraph neural networks (HyperGNN), graph autoencoder (GAE), and hyperbolic GNN. Table 5 shows the taxonomy of encoders used in existing work in detail.

Generally, in (C1) encoders, most methods represent each user p_i into two types of low-dimensional vectors (*i.e.*, embeddings) by employing a GNN encoder: p_i's interaction embedding $\mathbf{u}_i^I$ based on a U-I graph and p_i's social embedding $\mathbf{u}_i^S$ based on a U-U graph. Then, they aggregate them into one embedding $\mathbf{u}_i$ for the corresponding user p_i. In the meantime, they also obtain each item q_j's embedding $\mathbf{v}_j$ via another GNN encoder using a U-I graph.

It should be noted that some works employ only a single GNN encoder to obtain the two embeddings. In contrast, others use different GNN encoders for the embeddings of different node types (*i.e.*, users or items). For simplicity, however, we here explain the GNN encoders by generalizing them to any node type in the input graph.

5.1.1 ***GCN.*** Early works [23, 33, 46, 48, 50, 72, 74, 102, 107, 129] have focused on representing the user and item embeddings using GCN. Given a node n_i (*i.e.*, a user or an item) in the input graph (*i.e.*, U-I or U-U graphs), a n_i's embedding $\mathbf{e}_i^k$ in k-th layer is represented based on the embeddings

Table 5. Taxonomy of encoder architectures. It should be noted that some methods employ non-GNN encoders (*e.g.*, RNN, MLP, or just an embedding vector (Emb)) or no encoders (*i.e.*, –) to obtain embeddings.

User Social	**User Interest**	**Item Encoder**	**Models**
GANN	GANN	GANN	GraphRec [16], DiffNet++ [101], DiffNetLG [78], MutualRec [106], SR-HGNN [110], ASR [32], GNNTSR [58], GAT-NSR [64], SoRecGAT [89], PA-GAN [27], SOAP-VAE [90], GTN [26], PDARec [127], FeSoG [52], ESRF [117], SoHRML [51], FBNE [5], DISGCN [38], ME-LGN [62], SGA [53], GDSRec [6], DANSER [103], GraphRec+[17], SRAN [108], TAG [69], SDCRec [15], KConvGraph [87], HeteroGraphRec [71], SocialRippleNet [31], SCGRec [113], TGRec [1], GSFR [105], IGRec [10], DICER [18]
		Emb	SAN [30], HIDM [40], GLOW [36], GMAN [42], HSGNN [97], MrAPR [79]
		GCN	BFHAN [124], SHGCN [130]
	GRNN	GRNN	DGARec-R [82], SSRGNN [9], GNN-DSR [44]
		Emb	DREAM [80]
	RNN	GANN	FuseRec [65]
		RNN	SGHAN [98]
		Emb	SSDRec [111]
	-	-	GHSCF [2]
GCN	GCN	Emb	DiffNet [102], MEGCN [33], MPSR [46], HOSR [50],
		GCN	ATGCN [72], SENGR [74], SAGLG [48], GNN-SOR [23]
	GRNN	GRNN	GNNRec [45], MGSR [66]
		GCN	EGFRec [22]
		Emb	DGRec [81]
	RNN	GANN	MOHCN [96]
	MLP	Emb	MGNN [107]
LightGCN	LightGCN	LightGCN	SocialLGN [43], DcRec [100], APTE [126], EAGCN [99], CGL [123] SEPT [116], DSR [73], DESIGN [86]
		Emb	IDiffNet [41]
HetGNN	HetGNN	HetGNN	SeRec [8]
		GANN	DCAN [92]
HyperGNN	GANN	HyperGNN	MHCN [118], Motif-Res [83], DH-HGCN [24]
GAE	GAE	GAE	SIGA [47]
Hyperbolic	Hyperbolic	Hyperbolic	HyperSoRec [91]

of n_i's neighbors in $(k-1)$-th layer as follows:

$$\mathbf{e}_i^{(k)} = \sigma(\sum_{n_j \in \mathcal{N}_{n_i}} \mathbf{e}_j^{(k-1)} \mathbf{W}^{(k)}), \tag{1}$$

where σ and $\mathbf{W}^{(k)} \in \mathbb{R}^{d\times d}$ denote a non-linear activation function (*e*.g., ReLU) and a trainable transformation matrix, respectively. Also, $\mathcal{N}_{n_i}$ indicates a set of n_i's neighbors in the input graph. Here, some works take the self-connection of n_i into consideration by aggregating over the set $\mathcal{N}_{n_i} \cup \{n_i\}$. Most methods simply consider the n_i's embedding in the last K-th layer, $\mathbf{e}_i^{(K)}$, as its final embedding $\mathbf{z}_i$. Another variant is to aggregate n_i's embeddings from all layers, *i*.e., $\mathbf{z}_i = \sum_{k=1}^{K} \mathbf{e}_i^{(k)}$.

For instance, DiffNet [102] obtains a user p_i's social embedding $\mathbf{u}_i^S$ (resp. interaction embedding $\mathbf{u}_i^I$) by performing GCN with k-layers (resp. 1-layer) based on the U-U graph (resp. U-I graph). For each item q_j, it simply obtains q_j's embedding $\mathbf{v}_j$ based on its attributes without using a GNN encoder.

5.1.2 ***LightGCN***. It is well-known that non-linear activation and feature transformation in GCN encoders make the propagation step very complicated for training and scalability [25, 59]. Motivated by this, some works [41, 43, 73, 86, 99, 100, 116, 123, 126] have attempted to replace their GCN encoders with lightweight GCN [25], *i*.e.,

$$\mathbf{e}_i^{(k)} = \sum\nolimits_{n_j \in \mathcal{N}_{n_i}} \mathbf{e}_j^{(k-1)}. \tag{2}$$

It should be noted that LightGCN [25] has no non-linear activation function, no feature transformation, and no self-connection.

For instance, DcRec [100] obtains each user p_i's social embedding $\mathbf{u}_i^S$ via GCN, whereas obtaining the p_i's interaction embedding $\mathbf{u}_i^I$ and each item q_j's embedding $\mathbf{v}_j$ via the LightGCN encoder.

5.1.3 ***GANN***. The attention mechanism in graphs originated from the graph attention network (GAT) [88] and has already been successful in many applications, including recommender systems. Considering different weights from neighbor nodes in the input graph helps focus on important adjacent nodes while filtering out noises during the propagation process [88]. Therefore, almost all existing works on SocialRS have leveraged the attention mechanism in their GNN encoders [1, 2, 5, 6, 9, 10, 10, 15–18, 26, 27, 30–32, 36, 38, 40, 42, 44, 51–53, 58, 62, 64, 65, 69, 71, 78–80, 82, 87, 89, 90, 97, 98, 101, 103, 105, 106, 108, 110, 111, 113, 117, 124, 127, 130].

The common intuitions behind their design of the attention mechanism are: (1) each user's preferences for different items may differ, and (2) each user's influences on her social friends may differ. Based on such intuitions, many methods represent a node n_i's embedding in k-th layer by attentively aggregating the embeddings of n_i's neighbors in $(k-1)$-th layer as follows:

$$\mathbf{e}_i^{(k)} = \sigma(\sum\nolimits_{n_j \in \mathcal{N}_{n_i}} (\alpha_{ij} \cdot \mathbf{e}_j^{(k-1)})\mathbf{W}^{(k)}), \tag{3}$$

where α_{ij} indicates the attention weight of neighbor node n_j w.r.t n_i.

Now, we discuss how to compute the attention weights in existing works. Most methods, including DANSER [103] and SCGRec [113], typically use the concatenation-based graph attention as follows:

$$\alpha_{ij} = \frac{\exp(\text{MLP}[\mathbf{e}_i, \mathbf{e}_j])}{\sum_{n_k \in \mathcal{N}_{n_i}} \exp(\text{MLP}[\mathbf{e}_i, \mathbf{e}_j])}. \tag{4}$$

Also, other methods, including DICER [103] and MEGCN [33], use the similarity-based graph attention, which is another popular technique, *i*.e.,

$$\alpha_{ij} = \frac{\exp(\text{sim}(\mathbf{e}_i, \mathbf{e}_j))}{\sum_{n_k \in \mathcal{N}_{n_i}} \exp(\text{sim}(\mathbf{e}_i, \mathbf{e}_k))}, \tag{5}$$

where sim() denotes a similarity function such as cosine similarity and dot product.

5.1.4 ***HetGNN.*** The user-item interactions and user-user social relations can be regarded as the users' heterogeneous relationships, *i.e.*, a user's preferences on items and his/her friendship. In this sense, a few methods [8, 92] have attempted to model the inputs as a heterogeneous graph and then design the HetGNN encoders for learning user and item embeddings, *i.e.*,

$$\mathbf{e}_i^{(k)} = \sigma(\sum\nolimits_{n_j \in \mathcal{N}_{n_i}} \mathbf{e}_j^{(k-1)} \mathbf{W}_{v_{ij}}^{(k)}), \tag{6}$$

where v_{ij} indicates the type of relation between n_i and n_j. As a result, the HetGNN encoder employs different transformation matrices according to the relations between two nodes.

For instance, SeRec [8] defines four types of directed edges (*i.e.*, user-user edges, user-item edges, item-user edges, and item-item edges), constructing a heterogeneous graph based on the above edges. Then, it obtains each user p_i's embedding $\mathbf{u}_i$ and each item q_j's embedding $\mathbf{v}_j$ via the HetGNN encoder.

5.1.5 ***GRNN.*** The sequential behaviors of users when they interact with items reflect the evolution of their preferences of items over time. For this reason, time-aware recommender systems have attracted increasing attention in recent years [93]. Such temporal interactions are often divided into multiple user sessions and modeled as session-based SocialRS. Multiple works [9, 22, 44, 45, 66, 81, 82, 96, 107] have attempted to model dynamic user interests through session-based or temporal SocialRS. These models leverage the GRNN encoders to capture these time-evolving interests.

Suppose each user p interacts with items in a given sequence $\mathcal{S}_p$. Consequently, one can create a sequence of interactions for each item q as $\mathcal{S}_q$, consisting of users that rate item q in a temporal sequence. In general, the temporal sequence is denoted for node n_i as $\mathcal{S}_{n_i} = \{n_1^i, n_2^i, \cdots, n_K^i\}$. Note that session-based encoders would divide $\mathcal{S}_{n_i}$ into multiple sessions $\mathcal{S}_{n_i}^t$ and encode each session separately. The GRNN encoder for node n_i can be then generalized as:

$$\mathbf{e}_i = \mathrm{GRNN}(\mathcal{S}_{n_i}, \mathcal{N}_{n_i}), \tag{7}$$

where GRNN is a combination of RNN and GNN modules. In particular, one can obtain dynamic user interests and item embeddings through a long short-term memory (LSTM) [68, 120] unit, *i.e.*,

$$\begin{aligned}
\mathbf{x}_i^{(k)} &= \sigma(\mathbf{W}_x[\mathbf{h}_i^{(k-1)}, \mathbf{n}_k^i] + b_x), \\
\mathbf{f}_i^{(k)} &= \sigma(\mathbf{W}_s[\mathbf{h}_i^{(k-1)}, \mathbf{n}_k^i] + b_s), \\
\mathbf{o}_i^{(k)} &= \sigma(\mathbf{W}_o[\mathbf{h}_i^{(k-1)}, \mathbf{n}_k^i] + b_o), \\
\tilde{\mathbf{c}}_i^{(k)} &= \tanh(\mathbf{W}_c[\mathbf{h}_j^{(k-1)}, \mathbf{n}_k^i] + b_c), \\
\mathbf{c}_i^{(k)} &= \mathbf{f}_i^{(k)} \odot \mathbf{c}_i^{(k-1)} + \mathbf{x}_i^{(k)} \odot \tilde{\mathbf{c}}_i^{(k)}, \\
\mathbf{h}_i^{(k)} &= \mathbf{o}_i^{(k)} \odot \tanh(\mathbf{c}_i^{(k)}).
\end{aligned} \tag{8}$$

Then, the node embedding $\mathbf{e}_i$ is obtained using a GNN module such as GANN and GCN (as discussed below). In general, one can obtain

$$\mathbf{e}_i = \mathrm{GNN}(\mathbf{h}_i^{(K)}, \{\mathbf{h}_j^{(K)} : n_j \in \mathcal{N}_{n_i} \cup \{n_i\}\}). \tag{9}$$

For instance, DREAM [80] obtains each user p_i's embedding within each session using the GRNN encoder as above. It uses a Relational GAT module for the GNN layer to aggregate information from its social neighbors. Meanwhile, item embeddings $\mathbf{v}_j$ for item q_j is obtained using a simple embedding layer.

5.1.6 ***HyperGNN***. Most GNN encoders, as mentioned above, learn pairwise connectivity between two nodes. However, more-complicated connections can be captured by jointly using user-item relations with user-user edges and/or using higher-order social relations. For instance, triangular structures, including two users and their co-rated items, are a common motif. To leverage such high-order relations, some works [24, 83, 118] have attempted to model the inputs as a hypergraph and then design the HyperGNN encoders for learning user and item embeddings.

Let $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ denotes a hypergraph where $\mathcal{N}$ and $\mathcal{E}$ indicate sets of nodes and hyperedges, respectively. Each hyperedge connects any number of nodes. The hypergraph $\mathcal{G}$ can be denoted by an incident matrix $\mathbf{H}_{\mathcal{G}} \in \mathbb{R}^{x \times y}$, where x and y indicate the numbers of nodes and hyperedges, respectively. In $\mathbf{H}_{\mathcal{G}}$, $h(n, e) = 1$ if $n \in e$, otherwise 0. Also, $\mathbf{D}_{\mathcal{N}}$ and $\mathbf{D}_{\mathcal{E}}$ denote the diagonal matrices of the node and hyperedge degrees, respectively. In this case, each layer of the HyperGNN encoder is defined as:

$$\mathbf{E}^{(k)} = \mathbf{D}_{\mathcal{N}}^{-\frac{1}{2}} \mathbf{H}_{\mathcal{G}} \mathbf{D}_{\mathcal{E}}^{-1} \mathbf{H}_{\mathcal{G}}^{\top} \mathbf{D}_{\mathcal{N}}^{-\frac{1}{2}} \mathbf{E}^{(k-1)}. \tag{10}$$

We note that HyperGNN-based SocialRS methods [24, 118] remove non-linear activation and feature transformation as in the LightGCN encoder.

For instance, MHCN [118] designs three types of triangular motifs, constructing three incidence matrices, each representing a hypergraph induced by each motif. Then, it obtains each user p_i's embedding $\mathbf{u}_i$ via the multi-type HyperGNN encoders while obtaining each item q_j's embedding $\mathbf{v}_j$ via the GCN encoder.

5.1.7 ***Others***. Furthermore, we briefly describe the two encoders, GAE and hyperbolic GNN, each of which is employed by only one method. Liu et al. [47] pointed out that GCN is mainly suitable for semi-supervised learning tasks. On the other hand, they claimed that the goal of GAE coincides with that of the recommendation task, which is to minimize the reconstruction error of input and output [47]. For this reason, they proposed a SocialRS method, named SIGA, which employs GAE and is used for the rating prediction task.

Meanwhile, Wang et al. [91] pointed out that since existing methods usually learn the user and item embeddings in the Euclidean space, these methods fail to explore the latent hierarchical property in the data. For this reason, they proposed a SocialRS method, named HyperSoRec, which performs in the hyperbolic space because the exponential expansion of hyperbolic space helps preserve more-complex relationships between users and items [34].

5.2 Decoders

In this subsection, we group the decoders of GNN-based SocialRS into two categories: dot-product and multi-layer perceptron (MLP). Table 6 summarizes the taxonomy of these decoders.

5.2.1 ***Dot-product***. Many methods [5, 8–10, 15, 23, 24, 32, 33, 38, 40–43, 45–48, 50–53, 62, 72, 73, 78–80, 82, 83, 86, 91, 98–102, 108, 111, 113, 116–118, 123, 126, 127, 129, 130] simply predict a user p_i's preference $\hat{r}_{ij}$ on an item q_j via a dot product of their corresponding embeddings, *i.e.*,

$$\hat{r}_{ij} = \mathbf{u}_i \cdot \mathbf{v}_j^{\top}. \tag{11}$$

5.2.2 ***MLP***. More than half of the existing methods [1, 2, 6, 16–18, 22, 26, 27, 30, 31, 36, 44, 58, 64–66, 69, 71, 74, 80, 81, 87, 89, 90, 92, 96, 97, 103, 105–107, 110, 124] predict a user p_i's preference $\hat{r}_{ij}$ on an item q_j by employing MLP as follows:

$$\hat{r}_{ij} = \sigma_L(\mathbf{W}_L^{\top}(\sigma_{L-1}(\ldots\sigma_2(\mathbf{W}_2^{\top} \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_j \end{bmatrix} + \mathbf{b}_2)\ldots)) + \mathbf{b}_L, \tag{12}$$

where $\mathbf{W}_i$, $\mathbf{b}_i$, and σ_i denote the weight matrix, bias vector, and activation function for i-th layer's perceptron, respectively.

Table 6. Taxonomy of decoder architectures

Decoders	Models
Dot-product	DiffNet [102], DiffNet++ [101], DiffNetLG [78], MEGCN [33], ASR [32], ATGCN [72], GNN-SOR [23], DGARec [82], SAGLG [48], HIDM [40], SocialLGN [43], GMAN [42], MPSR [46], DREAM [80], SHGCN [130], SCGRec [113], PDARec [127], MHCN [118], SEPT [116], DcRec [100], DH-HGCN [24], SSDRec [111], SeRec [8], Motif-Res [83], APTE [126], EAGCN [99], HOSR [50], FeSoG [52], IGRec [10], ESRF [117], SDCRec [15], SoHRML [51], HyperSoRec [91], DSR [73], DESIGN [86], SRAN [108], IDiffNet [41], CGL [123], FBNE [5], MrAPR [79], GNNRec [45], SGHAN [98], SSRGNN [9], DISGCN [38], ME-LGN [62], SGA [53], SIGA [47]
MLP	GraphRec [16], GraphRec+ [17], DICER [18], SAN [30], KConvGraphRec [87], EGFRec [22], FuseRec [65], GNNTSR [58], GAT-NSR [64], TGRec [1], SoRecGAT [89], PA-GAN [27], GHSCF [2], HeteroGraphRec [71], GLOW [36], BFHAN [124], MGSR [66], GTN [26], MGNN [107], SR-HGNN [110], DANSER [103], MutualRec [106], GSFR [105], SOAP-VAE [90], SENGR [74], MOHCN [96], DCAN [92], TAG [69], HSGNN [97], Social-RippleNet [31], GNN-DSR [44], GDSRec [6], DGRec [81], DREAM [80]

5.3 Loss Functions

In this subsection, we first group the primary loss functions of GNN-based SocialRS into 4 categories: Bayesian personalized ranking (BPR) [70], mean squared error (MSE), cross-entropy (CE), and hinge loss. In addition, we found that some works additionally employ auxiliary loss functions. Thus, we further group these loss functions into 8 categories: social link prediction (LP) loss, self-supervised loss (SSL), group-based loss, adversarial (Adv) loss, path-based loss, knowledge distillation (KD) loss, sentiment-aware loss, and policy-network-based (Policy Net) loss. Table 7 summarizes the taxonomy of loss functions used in existing work.

5.3.1 ***Primary Loss Functions.*** Different primary loss functions are employed depending on whether the methods focus on explicit or implicit feedback.

MSE Loss. For the methods that focus on explicit feedback (*e.g.*, star ratings) of users, most of them [1, 2, 6, 16, 17, 23, 26, 27, 30, 31, 42, 44, 52, 58, 64, 66, 69, 71, 74, 82, 87, 96, 99, 103, 110, 126, 127] learn user and item embeddings via the MSE-based loss function $\mathcal{L}_{MSE}$, which is defined as follows:

$$\mathcal{L}_{MSE} = \sum_{p_i \in \mathcal{U}} \sum_{q_j \in \mathcal{N}_{p_i}} (\hat{r}_{ij} - r_{ij})^2, \tag{13}$$

where r_{ij} indicates p_i's real rating score on q_j. That is, the embeddings of p_i and q_j are learned, aiming at minimizing the differences between p_i's real and predicted scores, *i.e.*, r_{ij} and $\hat{r}_{ij}$, for q_j.

BPR Loss. For the methods that focus on implicit feedback (*e.g.*, click or browsing history) of users, most of them [10, 24, 32, 33, 36, 38, 40, 41, 43, 46, 48, 50, 51, 53, 62, 72, 73, 78, 79, 83, 97, 100–102, 105–108, 113, 116–118, 123, 130] learn user and item embeddings via the BPR-based loss

Table 7. Taxonomy of loss functions

Loss Functions			Models
Primary Objectives	**MSE**		GraphRec [16], GNNTSR [58], GAT-NSR [64], TGRec [1], PA-GAN [27], GHSCF [2], GraphRec+ [17], GTN [26], PDARec [127], GNN-SOR [23], DANSER [103], SAN [30], KConvGraphRec [87], HeteroGraphRec [71], GMAN [42], SR-HGNN [110], DGARec-R [82], MGSR [66], APTE [126], EAGCN [99], FeSoG [52], SENGR [74], MOHCN [96], TAG [69], Social-RippleNet [31], GNN-DSR [44], GDSRec [6]
	BPR		ASR [32], SAGLG [48], MGNN [107], HIDM [40], SocialLGN [43], MPSR [46], SHGCN [130], MutualRec [106], ATGCN [72], DiffNet [102], DiffNet++ [101], DiffNetLG [78], MEGCN [33], GLOW [36], SCGRec [113], SEPT [116], DcRec [100], MHCN [118], DH-HGCN [24], GSFR [105], HOSR [50], IGRec [10], ESRF [117], SoHRML [51], DSR [73], SRAN [108], IDiffNet [41], CGL [123], MrAPR [79], DISGCN [38], HSGNN [97], ME-LGN [62], SGA [53], Motif-Res [83]
	CE		DICER [18], SoRecGAT [89], DANSER [103], BFHAN [124], EGFRec [22], FuseRec [65], SeRec [8], SSDRec [111], DESIGN [86], FBNE [5], GNNRec [45], DCAN [92], SOAP-VAE [90] SGHAN [98], SSRGNN [9], GDSRec [6], SIGA [47], DGRec [81], DREAM [80]
	Hinge		HyperSoRec [91]
Auxiliary Objectives	**Social LP**		MGNN [107], MutualRec [106], SR-HGNN [110], APTE [126], SoHRML [51], FBNE [5]
	SSL	**Social**	SEPT [116], DcRec [100]
		Interaction	SDCRec [15], CGL [123], DISGCN [38], DCAN [92], DcRec [100]
		Motif	Motif-Res [83], MHCN [118]
	Group		GLOW [36], GMAN [42]
	Adv		ESRF [117]
	Path		SPEX [37]
	KD		DESIGN [86]
	Sentiment		SENGR [74]
	Policy Net.		DANSER [103]

function $\mathcal{L}_{BPR}$, which is defined as follows:

$$\mathcal{L}_{BPR} = -\sum_{p_i \in \mathcal{U}} \sum_{q_j \in \mathcal{N}_{p_i}} \sum_{q_k \in \mathcal{N}_{p_i} \setminus \mathcal{I}} \log \sigma(\hat{r}_{ij} - \hat{r}_{ik}), \tag{14}$$

where $\mathcal{U}$ and $\mathcal{N}_{n_i}$ denote a set of users and a set of items rated by p_i, respectively. $\hat{r}_{ij}$ and $\hat{r}_{ik}$ indicate p_i's preference on the rated item q_j and the (randomly-sampled) unrated item q_k, respectively. Also, σ indicates the sigmoid function. That is, the embeddings of p_i, q_j, and q_k are learned based on the intuition that p_i's preference $\hat{r}_{ij}$ on q_j is likely to be higher than p_i's preference $\hat{r}_{ik}$ on q_k.

CE Loss. Several methods [5, 6, 8, 9, 18, 22, 37, 45, 47, 65, 80, 81, 86, 89, 90, 92, 98, 103, 111, 124, 129] for implicit feedback learn user and item embeddings via the CE-based loss function $\mathcal{L}_{CE}$, which is defined as follows:

$$\mathcal{L}_{CE} = -\sum_{p_i \in \mathcal{U}} \sum_{q_j \in \mathcal{I}} r_{ij} \log(\hat{r}_{ij}) + (1 - r_{ij}) \log(1 - \hat{r}_{ij}), \tag{15}$$

where $\mathcal{I}$ indicates a set of items. It should be noted that $r_{ij} = 1$ if $q_j \in \mathcal{N}_{p_i}$, otherwise $r_{ij} = 0$. That is, the embeddings of p_i and q_j are learned, aiming at maximizing p_i's preferences on his/her rated items while minimizing p_i's preferences on his/her unrated items.

Hinge Loss. A method [91] for implicit feedback learns user and item embeddings via the hinge loss function $\mathcal{L}_{Hinge}$, which is defined as follows:

$$\mathcal{L}_{Hinge} = \sum_{p_i \in \mathcal{U}} \sum_{q_j \in \mathcal{N}_{p_i}} \sum_{q_k \in \mathcal{N}_{p_i} \setminus \mathcal{I}} \max(0, \lambda + (\hat{r}_{ij})^2 - (\hat{r}_{ik})^2), \tag{16}$$

where λ indicates the safety margin size. That is, the embeddings of p_i, q_j, and q_k are learned, aiming at ensuring that p_i's preferences on his/her rated items q_j are higher than those on his/her unrated items q_k at least by a margin of λ.

5.3.2 ***Auxiliary Loss Functions.*** Here, we discuss the auxiliary loss functions used by GNN-based SocialRS methods.

Social Link Prediction (LP) Loss. It should be noted that the primary objectives of the existing works focus on reconstructing the input U-I rating graph. Along with this, papers like MGNN [107], MutualRec [106], and SR-HGNN [110] learn the BPR-based social LP loss that aims at reconstructing the input U-U social graph. Through this method, user embeddings can be informed further to reconstruct the social relations, which allows them to better capture the social network structure that is essential for the more-effective social recommendation.

Self-Supervised Loss (SSL). SSL originated in image and text domains to address the deficiency of labeled data [49]. The basic idea of SSL is to assign labels for unlabeled data and exploit them additionally in the training process. It is well-known that the data sparsity problem significantly affects the performance of recommender systems. Therefore, there has recently been a surge of interest in SSL for recommender systems [119].

Some GNN-based SocialRS methods [15, 38, 83, 92, 100, 100, 116, 118, 123] designed SSL, which is derived from U-U social and/or U-I rating graphs. In this survey, we categorized them as social SSL and interaction-based SSL depending on the graph type employed to design SSL. For the social SSL, SEPT [116] augments different views related to users with the U-U social graph, and designs two socially-aware encoders that aim at reconstructing the augmented views. It adopts the regime of tri-training [128], which operates on the augmented-views above for self-supervised signals. For the interaction-based SSL, SDCRec [15] samples two items among items rated by a user, which have the highest similarities to the user. Then, it additionally utilizes them as self-supervised signals.

On the other hand, Motif-Res [83] and MHCN [118] explore the motif information in graph structure so that such information can be utilized as self-supervised signals. For instance, MHCN [118] constructs multi-type hyperedges, which are instances of a set of triangular relations, and designs SSL by leveraging the hierarchy in the hypergraph structures. It aims at reflecting the user node's local and global high-order connectivity patterns in different hypergraphs [118].

Group-based Loss. GLOW [36] and GMAN [42] make use of the user groups. Based on the group information, both methods additionally design the group-based loss. They define the group-item interaction as indicating a set of users that has interacted with an item. Then, they represent each group's embedding by attentively aggregating the users' embeddings within the corresponding group. Finally, the user and item embeddings are learned via a group-based loss so that each group's preferences on items rated by users in the corresponding group are likely to be higher than those of their unrated items.

Others. We briefly discuss the other loss functions that are employed by only one method. Yu et al. [117] designed an adversarial mechanism to consider the fact that social relations are very sparse, noisy, and multi-faceted in real-world social networks. On the other hand, Li et al. [37] pointed out that existing SocialRS methods fail to distinguish social influence from social homophily. To address this limitation, they designed an auxiliary loss function that models and captures the rich information conveyed by the formation of social homophily [37]. Furthermore, Tao et al. [86] leveraged the knowledge distillation (KD) technique into the social recommendation to address the overfitting problem of existing methods. Shi et al. [74] incorporated both sentiment information derived from reviews and interaction information captured by the GNN encoder. To this end, they designed an auxiliary loss function that captures different sentimental aspects of items from reviews [74]. Lastly, Wu et al. [103] designed a policy-based loss function based on a contextual multi-armed bandit [4], which dynamically weighs different social effects, *i.*e., social homophily, social influence, item-to-item homophily, and item-to-item influence.

6 EXPERIMENTAL SETUP

In this section, we discuss the experimental setup of GNN-based SocialRS methods. Specifically, we review 17 benchmark datasets and 8 evaluation metrics, that are widely used in GNN-based SocialRS methods.

6.1 Benchmark Datasets

We summarize the datasets widely used by existing GNN-based SocialRS methods in Table 8. These datasets come from 8 different application domains: product, location, movie, image, music, bookmark, microblog, and miscellaneous. We present the statistics of each dataset, including the numbers of users, items, ratings, and social relations, and a list of papers using the corresponding dataset. Since several versions exist per dataset, we chose the version that includes the most significant number of rating information.

*6.1.1 **Product-related Datasets**.*

Epinions. This dataset is collected from a now-defunct consumer review site, Epinions. It contains 355.8K trust relations from 18.0K users and 764.3K ratings from 18.0K users on 261.6K products. Here, a trust relation between two users indicates that one user trusts a review of a product written by another user. For each rating, this dataset originally provides the product name, its category, the rating score in the range [1, 5], the timestamp that a user rated on an item, and the helpfulness of this rating. 37 GNN-based SocialRS methods reviewed in this survey used this dataset [1, 2, 6, 15–18, 23, 26, 27, 37, 40, 44, 45, 51, 52, 58, 64, 65, 71, 73, 80, 82, 86, 87, 90, 91, 96, 101, 103, 105–107, 110, 124, 126, 127], which means the most popular in SocialRS.

Ciao. This dataset is collected from a consumer review site in the UK, Ciao (https://www.ciao.co.uk/). It contains 111.7K trust relations from 7.3K users and 283.3K ratings from 7.3K users on 104.9K products. The rating scale is from 1 to 5. This dataset was used from 34 GNN-based SocialRS methods reviewed in this survey [1, 2, 6, 15–18, 26, 27, 32, 40, 43, 44, 46, 47, 51, 52, 58, 65, 71, 73, 79, 82, 86, 87, 90, 91, 96, 99, 100, 105, 110, 124].

Beidan. This dataset is collected from a social e-commerce platform in China, Beidan (https://www.beidian.com/), which allows users' sharing behaviors. It includes 2.3K social relations from 2.8K users and 35.1K ratings from 2.8K users on 2.2K products. For each social relation, Li et al [38]

Table 8. Statistics of 17 publicly-available benchmark datasets. Dataset source is hyperlinked to each dataset name: datasets colored blue contain links for both user-item interactions and user-user relations, whereas the ones in red only contain one of the two due to unavailability of the other.

Domains	Datasets	# Users	# Items	# Ratings	# Social	Papers Used
Product	Epinions	18,088	261,649	764,352	355,813	[1, 2, 16–18, 23, 26, 27, 40, 58, 64, 90, 106, 107, 127], [15, 51, 52, 65, 71, 82, 87, 101, 103, 105, 110, 124, 126], [6, 37, 44, 45, 73, 80, 86, 91, 96]
	Ciao	7,317	104,975	283,319	111,781	[1, 2, 17, 27, 40, 43, 46, 58, 65, 71, 82, 87, 90, 110, 124], [6, 15, 16, 18, 26, 32, 47, 51, 52, 73, 86, 91, 99, 100, 105], [44, 79, 96]
	Beidan	2,841	2,298	35,146	2,367	[38]
	Beibei	24,827	16,864	1,667,320	197,590	[38]
Location	Yelp[1]	19,539	21,266	405,884	363,672	[23, 24, 30, 32, 33, 46, 78, 83, 89, 101, 102, 116, 118, 126], [5, 41, 50, 62, 69, 74, 79, 81, 86, 91, 96, 98, 99, 108, 123]
	Dianping	59,426	10,224	934,334	813,331	[100, 101]
	Gowalla	33,661	41,229	1,218,599	283,778	[8, 9, 39, 53, 72, 92, 98, 99, 117]
	Foursquare	39,302	45,595	3,627,093	304,030	[8, 9, 39, 92]
Movie	MovieLens	138,159	16,954	1,501,622	487,184	[5, 31, 48, 87]
	Flixster	58,470	38,076	3,619,736	667,313	[17, 23, 47, 51, 105–107]
	FilmTrust	1,508	2,071	35,497	1,853	[32, 47, 52, 64, 83, 127]
Image	Flickr	8,358	82,120	327,815	187,273	[30, 33, 41, 86, 99, 101, 102, 108, 123]
Music	Last.fm	1,892	17,632	92,834	25,434	[10, 43, 53, 62, 72, 87, 106, 116–118, 122]
Bookmark	Delicious	1,629	3,450	282,482	12,571	[8, 9, 22, 40, 44, 81, 92]
Microblog	Weibo	6,812	19,519	157,555	133,712	[37]
	Twitter	8,930	232,849	466,259	96,718	[37]
Miscellaneous	Douban	2,848	39,586	894,887	35,770	[1, 10, 22, 24, 45, 47, 50, 66, 71, 72, 81, 83, 90, 110], [62, 80, 116–118]

[1]Raw dataset is available at https://www.yelp.com/dataset/documentation/main.

collected when a user's friend clicks a link shared by the user that points to the information of a specific item. In this dataset, rating information does not provide explicit preference scores of users, rather containing implicit feedback only. This dataset was used in only one GNN-based SocialRS method [38].

Beibei. This dataset is collected from another social e-commerce platform in China, Beibei (https://www.beibei.com/). It is similar to Beidan but provides larger sizes of social relations and ratings. This dataset includes 197.5K social relations from 24.8K users and 1.6M ratings from 24.8K users on 16.8K products. For ratings, this dataset provides users' implicit feedback. This dataset was used in [38] only.

6.1.2 *Location-related Datasets*.

Yelp. This dataset is collected from a business review site, Yelp (https://www.yelp.com/). It contains 363.6K social relations from 19.5K users and 405.8K ratings from 19.5K users on 21.2K businesses. On Yelp, users can share their check-ins about local businesses (*e*.g., restaurants and home services) and express their experience through ratings in the range [0, 5]. Also, users can create social relations with other users. Each check-in contains a user, a timestamp, and a business (*i*.e., an item) that the user visited. 29 GNN-based SocialRS methods reviewed in this survey used this dataset [5, 23, 24, 30, 32, 33, 41, 46, 50, 62, 69, 74, 78, 79, 81, 83, 86, 89, 91, 96, 98, 99, 101, 102, 108, 116, 118, 123, 126].

Dianping. This dataset is collected from a local restaurant search and review platform in China, Dianping (https://www.dianping.com/). It contains 813.3K social relations from 59.4K users and 934.3K ratings from 59.4K users on 10.2K restaurants. For ratings, each user can give scores in the range [1, 5]. This dataset was used in two GNN-based SocialRS methods [100, 101].

Gowalla. This dataset is collected from a location-based social networking site, Gowalla (https://www.gowalla.com/). It contains 283.7K friendship relations from 33.6K users and 1.2M ratings from 33.6K users on 41.2K locations. On Gowalla, users can share information about their locations by check-in and make friends based on the shared information. For ratings, this dataset provides users' implicit feedback. 9 GNN-based SocialRS methods used this dataset [8, 9, 39, 53, 72, 92, 98, 99, 117].

Foursquare. This dataset is collected from another location-based social networking site, Foursquare (https://foursquare.com/). It is similar to Gowalla but provides larger sizes of social relations and ratings. It contains 304.0K friendship relations from 39.3K users and 3.6M ratings from 39.3K users on 45.5K locations. For ratings, this dataset provides users' implicit feedback. This dataset was used in 4 GNN-based SocialRS methods [8, 9, 39, 92].

6.1.3 *Movie-related Datasets*.

MovieLens. This dataset is collected from GroupLens Research (https://grouplens.org/) for the purpose of recommendation research. It contains 487.1K social relations from 138.1K users and 1.5M ratings from 138.1K users on 16.9K movies. It should be noted that this dataset has different versions according to the size of the rating information. For the details, refer to https://grouplens.org/datasets/movielens/. Since the original MovieLens datasets do not contain users' social relations, methods using this dataset built social relations by calculating the similarities between users. This dataset was used in 4 GNN-based SocialRS methods [5, 31, 48, 87].

Flixster. This dataset is collected from a movie review site, Flixster (https://www.flixster.com/). It contains 667.3K friendship relations from 58.4K users and 3.6M ratings from 58.4K users on 38.0K movies. On Flixster, users can add other users to their friend lists and express their preferences for movies. The rating values are 10 discrete numbers in the range [0.5, 5]. We found that 7 GNN-based SocialRS methods used this dataset [17, 23, 47, 51, 105–107].

FilmTrust. This dataset is collected from another (now-defunct) movie review site, FilmTrust. It is similar to Flixster but provides smaller sizes of social relations and ratings. It contains 1.8K friendship relations from 1.5K users and 35.4K ratings from 1.5K users on 2.0K movies. The rating scale is from 1 to 5. This dataset was used in 6 GNN-based SocialRS methods [32, 47, 52, 64, 83, 127].

6.1.4 *Image-related Dataset*.

Flickr. This dataset is collected from a who-trust-whom online image-based social sharing platform, Flickr (https://www.flickr.com/). It contains 187.2K follow relations from 8.3K users and 327.8K ratings from 8.3K users on 82.1K images. On Flickr, users can follow other users and share their preferences for images with their followers. For ratings, this dataset provides users' implicit feedback. Also, we found that 9 GNN-based SocialRS methods used this dataset [30, 33, 41, 86, 99, 101, 102, 108, 123].

6.1.5 *Music-related Dataset*.

Last.fm. This dataset is collected from a social music platform, Lat.fm (https://www.last.fm/). It contains 25.4K social relations from 1.8K users and 92.8K ratings from 1.8K users on 17.6K music artists. Each rating indicates that one user listened to an artist's music, *i.e.*, implicit feedback. On Lat.fm, users can make friend relations based on their preferences for artists. This dataset was used in 11 GNN-based SocialRS methods [10, 43, 53, 62, 72, 87, 106, 116–118, 122].

6.1.6 *Bookmark-related Dataset*.

Delicious. This dataset is collected from a social bookmarking system, Delicious (https://del.icio.us/). It contains 12.5K social relations from 1.6K users and 282.4K ratings from 1.6K users on 3.4K tags. On Delicious, users can bookmark URLs (*i.e.*, implicit feedback) and also assign a variety of semantic tags to bookmarks. Also, they can have social relations with other users having mutual bookmarks or tags. This dataset was used in 7 GNN-based SocialRS methods [8, 9, 22, 40, 44, 81, 92].

6.1.7 *Microblog-related Datasets*.

Weibo. This dataset is collected from a social microblog site in China, Weibo (https://weibo.com/). It contains 133.7K social relations from 6.8K users and 157.5K ratings from 6.8K users on 19.5K blogs. On Weibo, users can post microblogs (*i.e.*, implicit feedback) and retweet other users' blogs. Based on such retweeting behavior, Li et al. [37] collected social relations between users. Specifically, if a user has retweeted a microblog from another user, a social relation between the two users is created. This dataset was used in only one GNN-based SocialRS method [37].

Twitter. This dataset is collected from another social microblog site, Twitter (https://twitter.com/). It is similar to Weibo and contains 96.7K social relations from 8.3K users and 466.2K ratings from 8.9K users on 232.8K blogs. Li et al. [37] collected social relations between two users if a user retweets or replies to a tweet from another user. This dataset was used in [37] only.

6.1.8 *Miscellaneous*.

Douban. This dataset is collected from a social platform in China, Douban (https://douban.com/). It contains 35.7K social relations from 2.8K users and 894.8K ratings from 2.8K users on 39.5K items of different categories (*e.g.*, books, movies, movies, and so on). For ratings, this dataset provides users' implicit feedback. This dataset was used in 19 GNN-based SocialRS methods [1, 10, 22, 24, 45, 47, 50, 62, 66, 71, 72, 80, 81, 83, 90, 110, 116–118]. However, it should be noted that most methods using this dataset split users' ratings according to the item categories and then use those of some categories only, *e.g.*, Douban-Movie and Douban-Book.

6.2 Evaluation Metrics

6.2.1 ***Rating Prediction Task*.** The methods that focus on explicit feedback aim to minimize the errors of the rating prediction task. To evaluate the performance of this task, they use the following metrics: root mean squared error (RMSE) and mean absolute error (MAE). Specifically, MAE calculates the average error, the difference between the predicted and actual ratings, while

RMSE emphasizes larger errors. Both metrics are computed as follows:

$$
\begin{aligned}
MAE &= \frac{1}{M} \sum_{p_i \in \mathcal{U}} \sum_{q_j \in \mathcal{N}_{p_i}} |\hat{r}_{ij} - r_{ij}|, \\
RMSE &= \sqrt{\frac{1}{M} \sum_{p_i \in \mathcal{U}} \sum_{q_j \in \mathcal{N}_{p_i}} (\hat{r}_{ij} - r_{ij})^2},
\end{aligned}
\tag{17}
$$

where M indicates the number of ratings. Also, $\mathcal{U}$ and $\mathcal{N}_{p_i}$ denote a set of users and a set of items rated by p_i, respectively. Lastly, r_{ij} and $\hat{r}_{ij}$ indicate a user p_i's actual and predicted ratings on an item q_j, respectively.

6.2.2 ***Top-N Recommendation Task.*** The methods that focus on implicit feedback aim to improve the accuracy of the top-N recommendation task. To evaluate the performance of this task, they use the following metrics: normalized discounted cumulative gain (NDCG) [29], mean reciprocal rank (MRR) [3], area under the ROC curve (AUC), F1 score, precision, and recall.

First, NDCG reflects the importance of ranked positions of items in a set $\mathcal{R}_{p_i}$ of N items that each method recommends to a user p_i. Let y_k represent a binary variable for k-th item i_k in $\mathcal{R}_{p_i}$, *i.e.*, $y_k \in \{0, 1\}$. y_k is set as 1 if $i_k \in \mathcal{R}_{p_i}$ and set as 0 otherwise. $\mathcal{N}_{p_i}$ denotes a set of items considered relevant to p_i (*i.e.*, ground truth). In this case, $\text{NDCG}_{p_i}@N$ is computed by:

$$
\begin{aligned}
\text{NDCG}_{p_i}@N &= \frac{\text{DCG}_{p_i}@N}{\text{IDCG}_{p_i}@N}, \\
\text{DCG}_{p_i}@N &= \sum_{k=1}^{N} \frac{2^{y_k} - 1}{\log_2 (k+1)},
\end{aligned}
\tag{18}
$$

where $\text{IDCG}_{p_i}@N$ is the ideal DCG at N, *i.e.*, for every item i_k in $\mathcal{R}_{p_i}$, y_k is set as 1.

Second, MRR reflects the average inversed rankings of the first relevant item i_k in $\mathcal{R}_{p_i}$. $\text{MRR}_{p_i}@N$ is computed by:

$$
\text{MRR}_{p_i}@N = \frac{1}{\text{rank}_{p_i}}, \tag{19}
$$

where rank_{p_i} refers to the rank position of the first relevant item in $\mathcal{N}_{p_i}$.

Third, AUC evaluates whether each method ranks a rated item higher than an unrated item. That is, AUC_{p_i} is computed by:

$$
\text{AUC}_{p_i} = \frac{\sum_{q_j \in \mathcal{N}_{p_i}} \sum_{q_k \in \mathcal{N}_{p_i} \setminus \mathcal{I}} I(\hat{r}_{ij} > \hat{r}_{ik})}{|\mathcal{N}_{p_i}||\mathcal{N}_{p_i} \setminus \mathcal{I}|}, \tag{20}
$$

where $I(\cdot)$ is the indicator function.

Lastly, F1 score considers precision and recall together by taking their harmonic mean:

$$
\text{F1}_{p_i}@N = 2 \cdot \frac{\text{Precision}_{p_i}@N \cdot \text{Recall}_{p_i}@N}{\text{Precision}_{p_i}@N + \text{Recall}_{p_i}@N}, \tag{21}
$$

$$
\begin{aligned}
\text{Precision}_{p_i}@N &= \frac{|\mathcal{N}_{p_i} \bigcap \mathcal{N}_{p_i}|}{|\mathcal{R}_{p_i}|}, \\
\text{Recall}_{p_i}@N &= \frac{|\mathcal{N}_{p_i} \bigcap \mathcal{R}_{p_i}|}{|\mathcal{N}_{p_i}|},
\end{aligned}
\tag{22}
$$

where $\text{Precision}_{p_i}@N$ and $\text{Recall}_{p_i}@N$ denote precision and recall at N, respectively.

7 FUTURE DIRECTIONS

In this section, we discuss the limitations of GNN-based SocialRS methods and present several future research directions.

7.1 Graph Augmentation in GNN-based SocialRS

An intrinsic challenge of GNN-based SocialRS methods lies in the *sparsity of the input data* (*i*.e., user-item interactions and user-user relations). To mitigate this problem, some GNN-based SocialRS methods [15, 38, 83, 92, 100, 100, 116, 118, 123] have explored more supervision signals from the input data so that such signals can be utilized as different views from the original graph structure. Although many graph augmentation techniques [13, 125] such as node/edge deletion and graph rewiring have been proposed recently in a machine learning area, existing GNN-based SocialRS methods only focus on adding edges between two users or between a user and an item [15, 38, 83, 92, 100, 100, 116, 118, 123]. Therefore, it is a promising direction to leverage extra self-supervision signals based on various augmentation techniques to learn user and item embeddings more efficiently and effectively.

7.2 Trustworthy GNN-based SocialRS

Existing GNN-based SocialRS methods have focused on improving their *accuracy* by only relying on users' past feedback. However, it is worth mentioning that there are other important "beyond accuracy" metrics, which we call trustworthiness[2] according to [121]. Motivated by the importance of such metrics, various trustworthy GNN architectures have been proposed to incorporate core aspects of trustworthiness, including robustness, explainability, privacy, and fairness, in the context of GNN encoders [121]. One GNN-based SocialRS method is proposed in this direction to specifically address the privacy issue [52]. In particular, Liu et al. [52] devised a framework that stores user privacy data only in local devices individually and analyzes them together via federated learning. Thus, developing trustworthy GNN-based SocialRS is a wide open for research. For example, consider robustness: bad actors may want to target certain products to certain users in a SocialRS setting; how robust would existing GNN-based SocialRS be against such attackers is an unanswered question and opens opportunities to create accurate as well as robust models.

7.3 Heterogeneity

In real-world graphs, nodes and their interactions are often multi-typed. Such graphs, which are called heterogeneous graphs, convey rich information such as heterogeneous attributes, meta-path structures, and temporal properties. Although HetGNN encoders have recently attracted attention in many domains (*e*.g., healthcare and cybersecurity) [95], there have been only a few attempts to leverage such heterogeneity in SocialRS [8, 92]. Therefore, designing a HetGNN-based SocialRS method remains an open question for the future.

7.4 Efficiency and Scalability

Most real-world graphs are too large and also grow rapidly. However, most GNN-based SocialRS methods are too complicated, thus facing difficulty scaling to such large-scale graphs. Some works have attempted to make more scalable versions of models, including SocialLGN [43], SEPT [116], and DcRec [100], have attempted to remove the non-linear activation function, feature transformation, and self-connection, whereas Tao et al. [86] leveraged the knowledge distillation (KD) technique

[2] "Trustworthy" is defined in the Oxford Dictionary as follows: an object or a person that you can rely on to be good, honest, sincere, etc [121].

into SocialRS. However, designing a highly scalable GNN architecture is an important problem that remains challenging to date.

8 CONCLUSIONS

Although there has been a surge of papers on developing GNN-based social recommendation methods, no survey paper existed that reviewed them thoroughly. Our work is the first systematic and comprehensive survey that studies 80 papers on GNN-based SocialRS, collected by following the PRISMA guidelines. We present a novel taxonomy of inputs and architectures for GNN-based SocialRS, thus, categorizing different methods developed over the years in this important topic. Through this survey, we hope to enable the researchers of this field to better position their works in the recent trend while forming a gateway for the new researchers to get introduced to this important and hot topic. We hope this survey helps readers to grasp recent trends in SocialRS and develop novel GNN-based SocialRS methods.

REFERENCES

[1] Ting Bai, Youjie Zhang, Bin Wu, and Jian-Yun Nie. 2020. Temporal Graph Neural Networks for Social Recommendation. In *Proceedings of IEEE International Conference on Big Data (BigData)*. 898–903.
[2] Zhongqin Bi, Lina Jing, Meijing Shan, Shuming Dou, and Shiyang Wang. 2021. Hierarchical social recommendation model based on a graph neural network. *Wireless Communications and Mobile Computing* 2021 (2021).
[3] John S. Breese, David Heckerman, and Carl Myers Kadie. 1998. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In *Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI)*. 43–52.
[4] Sébastien Bubeck and Nicolò Cesa-Bianchi. 2012. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends in Machine Learning* 5, 1 (2012), 1–122.
[5] Hongxu Chen, Hongzhi Yin, Tong Chen, Weiqing Wang, Xue Li, and Xia Hu. 2022. Social Boosted Recommendation With Folded Bipartite Network Embedding. *IEEE Transactions on Knowledge and Data Engineering* 34, 2 (2022), 914–926. https://doi.org/10.1109/TKDE.2020.2982878
[6] Jiajia Chen, Xin Xin, Xianfeng Liang, Xiangnan He, and Jun Liu. 2022. GDSRec: Graph-Based DecentralizedCollaborative Filtering for SocialRecommendation. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2022).
[7] Rui Chen, Qingyi Hua, Yan-shuo Chang, Bo Wang, Lei Zhang, and Xiangjie Kong. 2018. A Survey of Collaborative Filtering-Based Recommender Systems: From Traditional Methods to Hybrid Methods Based on Social Networks. *IEEE Access* 6 (2018), 64301–64320.
[8] Tianwen Chen and Raymond Chi-Wing Wong. 2021. An efficient and effective framework for session-based social recommendation. In *Proceedings of ACM International Conference on Web Search and Data Mining (WSDM)*. 400–408.
[9] Yan Chen, Wanhui Qian, Dongqin Liu, Mengdi Zhou, Yipeng Su, Jizhong Han, and Ruixuan Li. 2022. Your Social Circle Affects Your Interests: Social Influence Enhanced Session-Based Recommendation. In *Proceedings of International Conference on Computational Science (ICCS)*. 549–562.
[10] Yujin Chen, Jing Wang, Zhihao Wu, and Youfang Lin. 2022. Integrating user-Group relationships under interest similarity constraints for social recommendation. *Knowledge-Based Systems* (2022), 108921.
[11] ShuiGuang Deng, Longtao Huang, Guandong Xu, Xindong Wu, and Zhaohui Wu. 2017. On Deep Learning for Trust-Aware Recommendations in Social Networks. *IEEE Transactions on Neural Networks and Learning Systems* 28, 5 (2017), 1164–1177.
[12] Yue Deng. 2022. Recommender Systems Based on Graph Embedding Techniques: A Review. *IEEE Access* 10 (2022), 51587–51633.
[13] Kaize Ding, Zhe Xu, Hanghang Tong, and Huan Liu. 2022. Data Augmentation for Deep Graph Learning: A Survey. *ACM SIGKDD Explorations* (2022).
[14] Yingtong Dou, Hao Yang, and Xiaolong Deng. 2016. A Survey of Collaborative Filtering Algorithms for Social Recommender Systems. In *Proceedings of International Conference on Semantics, Knowledge and Grids (SKG)*. 40–46.
[15] Jing Du, Zesheng Ye, Lina Yao, Bin Guo, and Zhiwen Yu. 2022. Socially-aware Dual Contrastive Learning for Cold-Start Recommendation. In *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 1927–1932.
[16] 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 ACM Web Conference (WWW)*. 417–426.

[17] Wenqi Fan, Yao Ma, Qing Li, Jianping Wang, Guoyong Cai, Jiliang Tang, and Dawei Yin. 2022. A Graph Neural Network Framework for Social Recommendations. *IEEE Transactions on Knowledge and Data Engineering* 34, 5 (2022), 2033–2047. https://doi.org/10.1109/TKDE.2020.3008732
[18] Bairan Fu, Wenming Zhang, Guangneng Hu, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2021. Dual side deep context-aware modulation for social recommendation. In *Proceedings of the ACM Web Conference (WWW)*. 2524–2534.
[19] Chen Gao, Yu Zheng, Nian Li, Yinfeng Li, Yingrong Qin, Jinghua Piao, Yuhan Quan, Jianxin Chang, Depeng Jin, Xiangnan He, and Yong Li. 2022. A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions. *ACM Transactions on Recommender Systems (TORS)* (2022).
[20] Francisco García-Sánchez, Ricardo Colomo Palacios, and Rafael Valencia-García. 2020. A social-semantic recommender system for advertisements. *Information Processing and Management* 57, 2 (2020), 102153.
[21] Fabio Gasparetti, Giuseppe Sansonetti, and Alessandro Micarelli. 2021. Community detection in social recommender systems: a survey. *Applied Intelligence* 51, 6 (2021), 3975–3995.
[22] Pan Gu, Yuqiang Han, Wei Gao, Guandong Xu, and Jian Wu. 2021. Enhancing session-based social recommendation through item graph embedding and contextual friendship modeling. *Neurocomputing* 419 (2021), 190–202.
[23] Zhiwei Guo and Heng Wang. 2020. A deep graph neural network-based mechanism for social recommendations. *IEEE Transactions on Industrial Informatics* 17, 4 (2020), 2776–2783.
[24] Jiadi Han, Qian Tao, Yufei Tang, and Yuhan Xia. 2022. DH-HGCN: Dual Homogeneity Hypergraph Convolutional Network for Multiple Social Recommendations. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 2190–2194.
[25] 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 International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 639–648.
[26] Thi Linh Hoang, Tuan Dung Pham, and Viet Cuong Ta. 2021. Improving Graph Convolutional Networks with Transformer Layer in social-based items recommendation. In *Proceedings of International Conference on Knowledge and Systems Engineering (KSE)*. 1–6.
[27] Liyang Hou, Wenping Kong, Yali Gao, Yang Chen, and Xiaoyong Li. 2021. PA-GAN: Graph Attention Network for Preference-Aware Social Recommendation. In *Journal of Physics: Conference Series*, Vol. 1848. 012141.
[28] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of ACM Conference on Recommender Systems (RecSys)*. 135–142.
[29] Kalervo Järvelin and Jaana Kekäläinen. 2000. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 41–48.
[30] Nan Jiang, Li Gao, Fuxian Duan, Jie Wen, Tao Wan, and Honglong Chen. 2021. SAN: Attention-based social aggregation neural networks for recommendation system. *International Journal of Intelligent Systems* (2021).
[31] Wenbo Jiang and Yanrui Sun. 2022. Social-RippleNet: Jointly modeling of ripple net and social information for recommendation. *Applied Intelligence* (2022), 1–16.
[32] Yanbin Jiang, Huifang Ma, Yuhang Liu, Zhixin Li, and Liang Chang. 2021. Enhancing social recommendation via two-level graph attentional networks. *Neurocomputing* 449 (2021), 71–84.
[33] Bo Jin, Ke Cheng, Liang Zhang, Yanjie Fu, Minghao Yin, and Lu Jiang. 2020. Partial relationship aware influence diffusion via a multi-channel encoding scheme for social recommendation. In *Proceedings of ACM International Conference on Information & Knowledge Management (CIKM)*. 585–594.
[34] Dmitri V. Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguñá. 2010. Hyperbolic Geometry of Complex Networks. *Physical Review E* 82 (2010).
[35] Adit Krishnan, Hari Cheruvu, Cheng Tao, and Hari Sundaram. 2019. A modular adversarial approach to social recommendation. In *Proceedings of ACM International Conference on Information & Knowledge Management (CIKM)*. 1753–1762.
[36] Youfang Leng and Li Yu. 2022. Incorporating global and local social networks for group recommendations. *Pattern Recognition* 127 (2022), 108601.
[37] Hui Li, Lianyun Li, Guipeng Xv, Chen Lin, Ke Li, and Bingchuan Jiang. 2021. SPEX: A Generic Framework for Enhancing Neural Social Recommendation. *ACM Transactions on Information Systems (TOIS)* 40, 2 (2021), 1–33.
[38] Nian Li, Chen Gao, Depeng Jin, and Qingmin Liao. 2022. Disentangled Modeling of Social Homophily and Influence for Social Recommendation. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2022).
[39] Quan Li, Xinhua Xu, Xinghong Liu, and CHEN Qi. 2022. An Attention-Based Spatiotemporal GGNN for Next POI Recommendation. *IEEE Access* (2022).
[40] Yuan Li and Kedian Mu. 2020. Heterogeneous Information Diffusion Model for Social Recommendation. In *Proceedings of IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*. 184–191.

[41] Yuqiang Li, Zhilong Zhan, Huan Li, and Chun Liu. 2022. Interest-aware influence diffusion model for social recommendation. *Journal of Intelligent Information Systems* 58, 2 (2022), 363–377.

[42] Guoqiong Liao, Xiaobin Deng, Changxuan Wan, and Xiping Liu. 2022. Group event recommendation based on graph multi-head attention network combining explicit and implicit information. *Information Processing & Management* 59, 2 (2022), 102797.

[43] Jie Liao, Wei Zhou, Fengji Luo, Junhao Wen, Min Gao, Xiuhua Li, and Jun Zeng. 2022. SocialLGN: Light Graph Convolution Network for Social Recommendation. *Information Sciences* (2022).

[44] Junfa Lin, Siyuan Chen, and Jiahai Wang. 2022. Graph neural networks with dynamic and static representations for social recommendation. In *Proceedings of International Conference on Database Systems for Advanced Applications (DASFAA)*. 264–271.

[45] Chun Liu, Yuxiang Li, Hong Lin, and Chaojie Zhang. 2022. GNNRec: Gated graph neural network for session-based social recommendation model. *Journal of Intelligent Information Systems* (2022), 1–20.

[46] Hai Liu, Chao Zheng, Duantengchuan Li, Zhaoli Zhang, Ke Lin, Xiaoxuan Shen, Neal N Xiong, and Jiazhang Wang. 2022. Multi-perspective social recommendation method with graph representation learning. *Neurocomputing* 468 (2022), 469–481.

[47] Jinxin Liu, Yingyuan Xiao, Wenguang Zheng, and Ching-Hsien Hsu. 2022. SIGA: social influence modeling integrating graph autoencoder for rating prediction. *Applied Intelligence* (2022), 1–16.

[48] Shenghao Liu, Bang Wang, Xianjun Deng, and Laurence T Yang. 2021. Self-Attentive Graph Convolution Network with Latent Group Mining and Collaborative Filtering for Personalized Recommendation. *IEEE Transactions on Network Science and Engineering (TNSE)* (2021).

[49] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Zhaoyu Wang, Li Mian, Jing Zhang, and Jie Tang. 2021. Self-supervised Learning: Generative or Contrastive. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2021).

[50] Yang Liu, Liang Chen, Xiangnan He, Jiaying Peng, Zibin Zheng, and Jie Tang. 2022. Modelling High-Order Social Relations for Item Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 34, 9 (2022), 4385–4397. https://doi.org/10.1109/TKDE.2020.3039463

[51] Zhen Liu, Xiaodong Wang, Ying Ma, and Xinxin Yang. 2022. Relational metric learning with high-order neighborhood interactions for social recommendation. *Knowledge and Information Systems (KAIS)* 64, 6 (2022), 1525–1547.

[52] Zhiwei Liu, Liangwei Yang, Ziwei Fan, Hao Peng, and Philip S Yu. 2022. Federated social recommendation with graph neural network. *ACM Transactions on Intelligent Systems and Technology (TIST)* 13, 4 (2022), 1–24.

[53] Yuanwei Liufu and Hong Shen. 2021. Social Recommendation via Graph Attentive Aggregation. In *Proceedings of International Conference on Parallel and Distributed Computing: Applications and Technologies (PDCAT)*. 369–382.

[54] Hao Ma, Irwin King, and Michael R. Lyu. 2009. Learning to recommend with social trust ensemble. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 203–210.

[55] Hao Ma, Michael R. Lyu, and Irwin King. 2009. In *Proceedings of ACM Conference on Recommender Systems (RecSys)*. 189–196.

[56] Hao Ma, Haixuan Yang, Michael R. Lyu, and Irwin King. 2008. SoRec: social recommendation using probabilistic matrix factorization. In *Proceedings of ACM International Conference on Information & Knowledge Management (CIKM)*. 931–940.

[57] Hao Ma, Dengyong Zhou, Chao Liu, Michael R. Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *Proceedings of ACM International Conference on Web Search and Data Mining (WSDM)*. 287–296.

[58] Supriyo Mandal and Abyayananda Maiti. 2021. Graph Neural Networks for Heterogeneous Trust based Social Recommendation. In *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN)*. 1–8.

[59] Kelong Mao, Jieming Zhu, Xi Xiao, Biao Lu, Zhaowei Wang, and Xiuqiang He. 2021. UltraGCN: Ultra Simplification of Graph Convolutional Networks for Recommendation. In *Proceedings of ACM International Conference on Information & Knowledge Management (CIKM)*. 1253–1262.

[60] Peter V Marsden and Noah E Friedkin. 1993. Network studies of social influence. *Sociological Methods & Research* 22, 1 (1993), 127–151.

[61] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a feather: Homophily in social networks. *Annual Review of Sociology* (2001), 415–444.

[62] Hang Miao, Anchen Li, and Bo Yang. 2022. Meta-path Enhanced Lightweight Graph Neural Network for Social Recommendation. In *Proceedings of International Conference on Database Systems for Advanced Applications (DASFAA)*. 134–149.

[63] David Moher, Alessandro Liberati, Jennifer Tetzlaff, Douglas G Altman, and PRISMA Group*. 2009. Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement. *Annals of Internal Medicine* 151, 4 (2009), 264–269.

[64] Nan Mu, Daren Zha, Yuanye He, and Zhihao Tang. 2019. Graph attention networks for neural social recommendation. In *Proceedings of IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*. 1320–1327.

[65] Kanika Narang, Yitong Song, Alexander Schwing, and Hari Sundaram. 2021. FuseRec: fusing user and item homophily modeling with temporal recommender systems. *Data Mining and Knowledge Discovery (DMKD)* 35, 3 (2021), 837–862.
[66] Yong Niu, Xing Xing, Mindong Xin, Qiuyang Han, and Zhichun Jia. 2021. Multi-preference Social Recommendation of Users Based on Graph Neural Network. In *Proceedings of International Conference on Intelligent Computing, Automation and Applications (ICAA)*. 190–194.
[67] Alexis Papadimitriou, Panagiotis Symeonidis, and Yannis Manolopoulos. 2012. A generalized taxonomy of explanations styles for traditional and social recommender systems. *Data Mining and Knowledge Discovery (DMKD)* 24, 3 (2012), 555–583.
[68] Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. 2017. Cross-sentence n-ary relation extraction with graph lstms. *Transactions of the Association for Computational Linguistics (TACL)* 5 (2017), 101–115.
[69] Pengpeng Qiao, Zhiwei Zhang, Zhetao Li, Yuanxing Zhang, Kaigui Bian, Yanzhou Li, and Guoren Wang. 2022. TAG: Joint Triple-hierarchical Attention and GCN for Review-based Social Recommender System. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2022).
[70] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI)*. 452–461.
[71] Amirreza Salamat, Xiao Luo, and Ali Jafari. 2021. HeteroGraphRec: A heterogeneous graph-based neural networks for social recommendations. *Knowledge-Based Systems* 217 (2021), 106817.
[72] Dewen Seng, Binquan Li, Chenxuan Lai, and Jiayi Wang. 2021. Adaptive Learning User Implicit Trust Behavior Based on Graph Convolution Network. *IEEE Access* 9 (2021), 108363–108372.
[73] Xiao Sha, Zhu Sun, and Jie Zhang. 2021. Disentangling Multi-Facet Social Relations for Recommendation. *IEEE Transactions on Computational Social Systems* (2021).
[74] Liye Shi, Wen Wu, Wang Guo, Wenxin Hu, Jiayi Chen, Wei Zheng, and Liang He. 2022. SENGR: Sentiment-Enhanced Neural Graph Recommender. *Information Sciences* 589 (2022), 655–669.
[75] Jyoti Shokeen and Chhavi Rana. 2018. A review on the dynamics of social recommender systems. *International Journal of Web Engineering and Technology* 13, 3 (2018), 255–276.
[76] Jyoti Shokeen and Chhavi Rana. 2020. Social recommender systems: techniques, domains, metrics, datasets and future scope. *Journal of Intelligent Information Systems* 54, 3 (2020), 633–667.
[77] Jyoti Shokeen and Chhavi Rana. 2020. A study on features of social recommender systems. *Artificial Intelligence Review* 53, 2 (2020), 965–988.
[78] Changhao Song, Bo Wang, Qinxue Jiang, Yehua Zhang, Ruifang He, and Yuexian Hou. 2021. Social Recommendation with Implicit Social Influence. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 1788–1792.
[79] Hongtao Song, Feng Wang, Zhiqiang Ma, and Qilong Han. 2022. Multirelationship Aware Personalized Recommendation Model. In *Proceedings of International Conference of Pioneering Computer Scientists, Engineers and Educators (ICPCSEE)*. 123–136.
[80] Liqiang Song, Ye Bi, Mengqiu Yao, Zhenyu Wu, Jianming Wang, and Jing Xiao. 2020. Dream: A dynamic relation-aware model for social recommendation. In *Proceedings of ACM International Conference on Information & Knowledge Management (CIKM)*. 2225–2228.
[81] 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 ACM International Conference on Web Search and Data Mining (WSDM)*. 555–563.
[82] Hongji Sun, Lili Lin, and Riqing Chen. 2020. Social Recommendation based on Graph Neural Networks. In *Proceedings of IEEE International Conference on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*. 489–496.
[83] Yundong Sun, Dongjie Zhu, Haiwen Du, and Zhaoshuo Tian. 2022. Motifs-based Recommender System via Hypergraph Convolution and Contrastive Learning. *Neurocomputing* (2022).
[84] Jiliang Tang, Xia Hu, Huiji Gao, and Huan Liu. 2013. Exploiting Local and Global Social Context for Recommendation. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. 2712–2718.
[85] Jiliang Tang, Xia Hu, and Huan Liu. 2013. Social recommendation: a review. *Social Network Analysis and Mining* 3, 4 (2013), 1113–1133.
[86] Ye Tao, Ying Li, Su Zhang, Zhirong Hou, and Zhonghai Wu. 2022. Revisiting Graph based Social Recommendation: A Distillation Enhanced Social Graph Network. In *Proceedings of ACM Web Conference (WWW)*. 2830–2838.
[87] Dong Nguyen Tien and Hai Pham Van. 2020. Graph Neural Network Combined Knowledge Graph for Recommendation System. In *Proceedings of International Conference on Computational Data and Social Networks (CSoNet)*. 59–70.
[88] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. In *Proceedings of International Conference on Learning Representations (ICLR)*.

[89] M Vijaikumar, Shirish Shevade, and M Narasimha Murty. 2019. SoRecGAT: Leveraging graph attention mechanism for top-N social recommendation. In *Proceedings of Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD)*. 430–446.

[90] Joojo Walker, Fengli Zhang, Fan Zhou, and Ting Zhong. 2021. Social-trust-aware variational recommendation. *International Journal of Intelligent Systems* (2021).

[91] Hao Wang, Defu Lian, Hanghang Tong, Qi Liu, Zhenya Huang, and Enhong Chen. 2021. Hypersorec: Exploiting hyperbolic user and item representations with multiple aspects for social-aware recommendation. *ACM Transactions on Information Systems (TOIS)* 40, 2 (2021), 1–28.

[92] Liuyin Wang, Xianghong Xu, Kai Ouyang, Huanzhong Duan, Yanxiong Lu, and Hai-Tao Zheng. 2022. Self-Supervised Dual-Channel Attentive Network for Session-based Social Recommendation. In *Proceedings of IEEE International Conference on Data Engineering (ICDE)*. 2034–2045.

[93] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z Sheng, Mehmet A Orgun, and Defu Lian. 2021. A survey on session-based recommender systems. *ACM Computing Surveys (CSUR)* 54, 7 (2021), 1–38.

[94] Shoujin Wang, Liang Hu, Yan Wang, Xiangnan He, Quan Z. Sheng, Mehmet A. Orgun, Longbing Cao, Francesco Ricci, and Philip S. Yu. 2021. Graph Learning based Recommender Systems: A Review. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. 4644–4652.

[95] Xiao Wang, Deyu Bo, Chuan Shi, Shaohua Fan, Yanfang Ye, and Philip S. Yu. 2022. A Survey on Heterogeneous Graph Embedding: Methods, Techniques, Applications and Sources. *IEEE Transactions on Big Data* (2022).

[96] Yu Wang and Qilong Zhao. 2022. Multi-Order Hypergraph Convolutional Neural Network for Dynamic Social Recommendation System. *IEEE Access* 10 (2022), 87639–87649.

[97] Chunyu Wei, Yushun Fan, and Jia Zhang. 2022. High-order Social Graph Neural Network for Service Recommendation. *IEEE Transactions on Network and Service Management (TNSM)* (2022).

[98] Chunyu Wei, Yushun Fan, and Jia Zhang. 2022. Time-aware Service Recommendation with Social-powered Graph Hierarchical Attention Network. *IEEE Transactions on Services Computing* (2022).

[99] Bin Wu, Lihong Zhong, Lina Yao, and Yangdong Ye. 2022. EAGCN: An Efficient Adaptive Graph Convolutional Network for Item Recommendation in Social Internet of Things. *IEEE Internet of Things Journal* (2022).

[100] Jiahao Wu, Wenqi Fan, Jingfan Chen, Shengcai Liu, Qing Li, and Ke Tang. 2022. Disentangled Contrastive Learning for Social Recommendation. In *Proceedings of ACM International Conference on Information & Knowledge Management (CIKM)*. 4570–4574.

[101] Le Wu, Junwei Li, Peijie Sun, Richang Hong, Yong Ge, and Meng Wang. 2020. Diffnet++: A neural influence and interest diffusion network for social recommendation. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2020).

[102] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. 2019. A neural influence diffusion model for social recommendation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 235–244.

[103] 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 ACM Web Conference (WWW)*. 2091–2102.

[104] Shiwen Wu, Wentao Zhang, Fei Sun, and Bin Cui. 2022. Graph Neural Networks in Recommender Systems: A Survey. *ACM Computing Surveys (CSUR)* 37, 4 (2022).

[105] Xinyu Xiao, Junhao Wen, Wei Zhou, Fengji Luo, Min Gao, and Jun Zeng. 2022. Multi-interaction fusion collaborative filtering for social recommendation. *Expert Systems with Applications* (2022), 117610.

[106] Yang Xiao, Qingqi Pei, Tingting Xiao, Lina Yao, and Huan Liu. 2021. MutualRec: joint friend and item recommendations with mutualistic attentional graph neural networks. *Journal of Network and Computer Applications* 177 (2021), 102954.

[107] 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 Intelligent Systems* 35, 5 (2020), 7–17.

[108] Xiaojun Xie, Xihuang Zhang, Honggang Luo, and Tao Zhang. 2022. Similarity-based Multi-Relational Attention Network for Social Recommendation. In *Proceedings of International Conference on Computing and Artificial Intelligence (ICCAI)*. 307–317.

[109] Guandong Xu, Zhiang Wu, Yanchun Zhang, and Jie Cao. 2015. Social networking meets recommender systems: survey. *International Journal Social Network Mining* 2, 1 (2015), 64–100.

[110] Huance Xu, Chao Huang, Yong Xu, Lianghao Xia, Hao Xing, and Dawei Yin. 2020. Global context enhanced social recommendation with hierarchical graph neural networks. In *Proceedings of IEEE International Conference on Data Mining (ICDM)*. 701–710.

[111] Dengcheng Yan, Tianyi Tang, Wenxin Xie, Yiwen Zhang, and Qiang He. 2022. Session-based social and dependency-aware software recommendation. *Applied Soft Computing* 118 (2022), 108463.

[112] Bo Yang, Yu Lei, Dayou Liu, and Jiming Liu. 2013. Social Collaborative Filtering by Trust. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. 2747–2753.
[113] Liangwei Yang, Zhiwei Liu, Yu Wang, Chen Wang, Ziwei Fan, and Philip S. Yu. 2022. Large-scale Personalized Video Game Recommendation via Social-aware Contextualized Graph Neural Network. In *Proceedings of the ACM Web Conference (WWW)*. 3376–3386.
[114] Xiwang Yang, Yang Guo, Yong Liu, and Harald Steck. 2014. A survey of collaborative filtering based social recommender systems. *Computer Communications* 41 (2014), 1–10.
[115] Haochao Ying, Liang Chen, Yuwen Xiong, and Jian Wu. 2016. Collaborative Deep Ranking: A Hybrid Pair-Wise Recommendation Algorithm with Implicit Feedback. In *Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, Vol. 9652. 555–567.
[116] Junliang Yu, Hongzhi Yin, Min Gao, Xin Xia, Xiangliang Zhang, and Nguyen Quoc Viet Hung. 2021. Socially-aware self-supervised tri-training for recommendation. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD)*. 2084–2092.
[117] Junliang Yu, Hongzhi Yin, Jundong Li, Min Gao, Zi Huang, and Lizhen Cui. 2022. Enhancing Social Recommendation With Adversarial Graph Convolutional Networks. *IEEE Transactions on Knowledge and Data Engineering* 34, 8 (2022), 3727–3739. https://doi.org/10.1109/TKDE.2020.3033673
[118] Junliang Yu, Hongzhi Yin, Jundong Li, Qinyong Wang, Nguyen Quoc Viet Hung, and Xiangliang Zhang. 2021. Self-supervised multi-channel hypergraph convolutional network for social recommendation. In *Proceedings of the ACM Web Conference (WWW)*. 413–424.
[119] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Jundong Li, and Zi Huang. 2022. Self-Supervised Learning for Recommender Systems: A Survey. *arXiv:2203.15876* (2022).
[120] Victoria Zayats and Mari Ostendorf. 2018. Conversation modeling on Reddit using a graph-structured LSTM. *Transactions of the Association for Computational Linguistics (TACL)* 6 (2018), 121–132.
[121] He Zhang, Bang Wu, Xingliang Yuan, Shirui Pan, Hanghang Tong, and Jian Pei. 2022. Trustworthy Graph Neural Networks: Aspects, Methods and Trends. *arXiv:2205.07424* (2022).
[122] Lisa Zhang, Zhe Kang, Xiaoxin Sun, Hong Sun, Bangzuo Zhang, and Dongbing Pu. 2021. KCRec: Knowledge-aware representation Graph Convolutional Network for Recommendation. *Knowledge-Based Systems* 230 (2021), 107399.
[123] Yongshuai Zhang, Jiajin Huang, Mi Li, and Jian Yang. 2022. Contrastive Graph Learning for Social Recommendation. *Frontiers in Physics* (2022), 35.
[124] Minghao Zhao, Qilin Deng, Kai Wang, Runze Wu, Jianrong Tao, Changjie Fan, Liang Chen, and Peng Cui. 2021. Bilateral filtering graph convolutional network for multi-relational social recommendation in the power-law networks. *ACM Transactions on Information Systems (TOIS)* 40, 2 (2021), 1–24.
[125] Tong Zhao, Gang Liu, Stephan Günnemann, and Meng Jiang. 2022. Graph Data Augmentation for Graph Machine Learning: A Survey. *arXiv:2202.08871* (2022).
[126] Yan Zhen, Huan Liu, Meiyu Sun, Boran Yang, and Puning Zhang. 2022. Adaptive preference transfer for personalized IoT entity recommendation. *Pattern Recognition Letters* 162 (2022), 40–46.
[127] Li Zheng, Qun Liu, and Youmin Zhang. 2021. Social Recommendation Based on Preference Disentangle Aggregation. In *Proceedings of International Conference on Big Data and Information Analytics (BigDIA)*. 1–8.
[128] Zhi-Hua Zhou and Ming Li. 2005. Tri-Training: Exploiting Unlabeled Data Using Three Classifiers. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 17, 11 (2005), 1529–1541.
[129] Peng Zhu, Dawei Cheng, Siqiang Luo, Fangzhou Yang, Yifeng Luo, Weining Qian, and Aoying Zhou. 2022. SI-News: Integrating social information for news recommendation with attention-based graph convolutional network. *Neurocomputing* 494 (2022), 33–42.
[130] Zirui Zhu, Chen Gao, Xu Chen, Nian Li, Depeng Jin, and Yong Li. 2021. Inhomogeneous Social Recommendation with Hypergraph Convolutional Networks. In *Proceedings of IEEE International Conference on Data Engineering (ICDE)*.