MEGCF: Multimodal Entity Graph Collaborative Filtering for Personalized Recommendation

KANG LIU, FENG XUE, DAN GUO, LE WU, SHUJIE LI, and RICHANG HONG,
Hefei University of Technology, China

In most E-commerce platforms, whether the displayed items trigger the user's interest largely depends on their most eye-catching multimodal content. Consequently, increasing efforts focus on modeling multimodal user preference, and the pressing paradigm is to incorporate complete multimodal deep features of the items into the recommendation module. However, the existing studies **ignore the mismatch problem between multimodal feature extraction (MFE) and user interest modeling (UIM)**. That is, MFE and UIM have different emphases. Specifically, MFE is migrated from and adapted to upstream tasks such as image classification. In addition, it is mainly a content-oriented and non-personalized process, while UIM, with its greater focus on understanding user interaction, is essentially a user-oriented and personalized process. Therefore, the direct incorporation of MFE into UIM for purely user-oriented tasks, tends to introduce a large number of preference-independent multimodal noise and contaminate the embedding representations in UIM.

This paper aims at solving the mismatch problem between MFE and UIM, so as to generate high-quality embedding representations and better model multimodal user preferences. Towards this end, we develop a novel model, multimodal entity graph collaborative filtering, short for MEGCF. The UIM of the proposed model captures the semantic correlation between interactions and the features obtained from MFE, thus making a better match between MFE and UIM. More precisely, semantic-rich entities are first extracted from the multimodal data, since they are more relevant to user preferences than other multimodal information. These entities are then integrated into the user-item interaction graph. Afterwards, a symmetric linear **Graph Convolution Network (GCN)** module is constructed to perform message propagation over the graph, in order to capture both high-order semantic correlation and collaborative filtering signals. Finally, the sentiment information from the review data are used to fine-grainedly weight neighbor aggregation in the GCN, as it reflects the overall quality of the items, and therefore it is an important modality information related to user preferences. Extensive experiments demonstrate the effectiveness and rationality of MEGCF.[1]

CCS Concepts: • **Information systems** → **Recommender systems**; **Personalization;**

[1]We release the complete codes of MEGCF at https://github.com/hfutmars/MEGCF.

This work is supported in part by the Seventh Special Support Plan for Innovation and Entrepreneurship in Anhui Province, in part by the Anhui Provincial Major Science and Technology Project under Grant 202203a05020025, and in part by the National Natural Science Foundation of China under Grant 61876058.

Authors' addresses: K. Liu, D. Guo, L. Wu, and R. Hong, Hefei University of Technology, School of Computer Science and Information Engineering, Key Laboratory of Knowledge Engineering with Big Data, Intelligent Interconnected Systems Laboratory of Anhui Province, 485 Danxia Road, Hefei, Anhui Province, China, 230601; emails: kangliu1225@gmail.com, guodan@hfut.edu.cn, lewu.ustc@gmail.com, hongrc@hfut.edu.cn; F. Xue (corresponding author) and S. Li, Hefei University of Technology, School of Software, Key Laboratory of Knowledge Engineering with Big Data, Intelligent Interconnected Systems Laboratory of Anhui Province, 485 Danxia Road, Hefei, Anhui Province, China, 230601; emails: {feng.xue, lisjhfut}@hfut.edu.cn.

1046-8188/2023/03-ART30 $15.00
https://doi.org/10.1145/3544106

Additional Key Words and Phrases: Collaborative filtering, semantic correlation, multimodal user preference, multimodal semantic entity, collaborative signal, graph convolutional network, sentiment analysis

ACM Reference format:
Kang Liu, Feng Xue, Dan Guo, Le Wu, Shujie Li, and Richang Hong. 2023. MEGCF: Multimodal Entity Graph Collaborative Filtering for Personalized Recommendation. *ACM Trans. Inf. Syst.* 41, 2, Article 30 (March 2023), 27 pages.
https://doi.org/10.1145/3544106

1 INTRODUCTION

The personalized recommender algorithm plays a crucial role in many online services, such as E-commerce, content-sharing platform, and social media. **Collaborative Filtering (CF)** [26] is the most widely used recommender method, which assumes that there is a correlation signal between observed user-item pairs through collaborative relationships, and the signal enables accurate assessment of users' preference over items, which is referred to as CF signal in some studies [38, 39]. However, CF faces the challenges of sparsity and cold-start, that is, the inability to capture sufficient CF signals from sparse interactions to generate high-quality recommendations. Content-enriched recommender methods can efficiently alleviate this problem by drawing on additional information of users and items, such as demographic features [3], attributes of items [6], social relationships [47], knowledge graphs [35], and multimodal content (e.g., images, short videos, titles, reviews, *etc.*) of items [43] (referred to as multimodal recommender method), in order to enrich the representations of users and items. This work focuses on the research of multimodal recommender methods due to the following two considerations: (1) in most recommendation scenarios, multimodal information is the dominant presentation of the item and it directly engages with users. Therefore, it contains abundant user preference-related clues that differ from the collaborative relationships in the interactions; and (2) the recent success of video-sharing platforms, such as Tiktok and Kwai, bring increasing attention to extracting user preference over multimodal content.

The existing multimodal recommender efforts can be broadly categorized into two types of frameworks: **Separated Framework (SF)** and **End2end Framework (EF)**. They are both equipped with two main modules: **multimodal feature extraction (MFE)** and **user interest modeling (UIM)**. In the SF-based methods [8, 28, 43, 52], the MFE module uses a pre-trained network migrated from the upstream task to extract the full range of multimodal deep features. In addition, the UIM module incorporates these deep features into the user preference modeling. Compared with SF, EF-based methods [14, 15, 19, 30, 53] differ by fusing the MFE and UIM modules into an end2end framework and using interaction data to jointly optimize them. Note that a more comprehensive overview of these multimodal recommender methods is provided in Section 2.3.

Although the existing studies demonstrate the effectiveness of this schema, they ignore the mismatch problem between the MFE and UIM modules. Specifically, MFE, as a module tailored for upstream tasks, aims at mining multimodal deep features that are relevant to the specific upstream task but not to the user preferences. Therefore, it is a content-oriented process. On the contrary, UIM, as the core of recommendation task, is a user-oriented and personalized module which aims at collecting and then processing preference-related features. In general, MFE and UIM have completely different emphases. In other words, they are mismatched in user preference inference, thus limiting the positive impact of multimodal information. In addition, the features output by the MFE module contain a large amount of preference-independent noisy data, resulting in contamination of the embedding representations. Figure 1 shows a specific mismatch phenomenon between user preferences and **visual feature extraction (VFE)** module. That is, whether or not a user will purchase an item is highly related to the semantic-rich entities (*e.g.*, jacket, white hat, and jeans) in

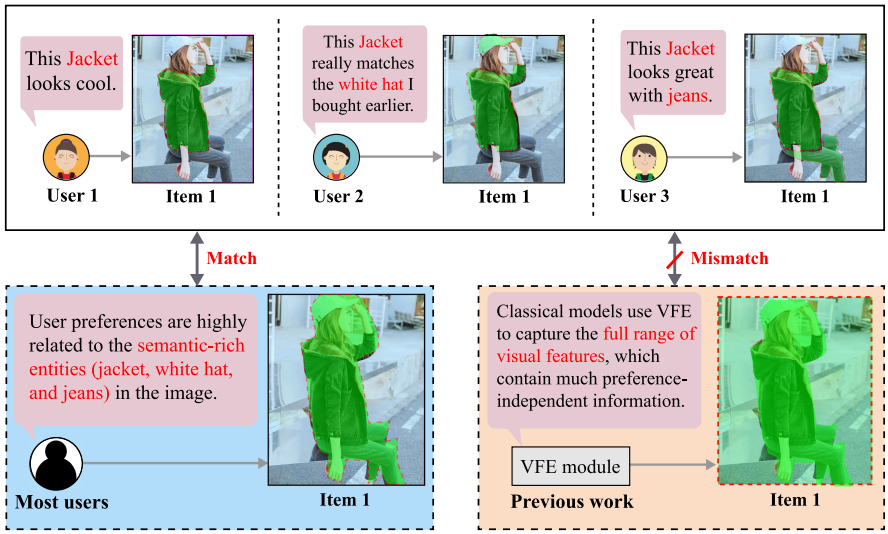

Fig. 1. An example to illustrate the mismatch between user preferences and the visual feature extraction (VFE) module when processing visual content, where *VFE* is generally a convolutional neural network pre-trained on a large-scale visual dataset; *User 1*, *User 2*, and *User 3* are users who interact with *Item 1* together.

the image of this item, while the VFE module captures the full range of visual features that contain a large number of preference-independent information (*e.g.*, background, brightness, and the relative position of the entities).

In order to solve the mismatch problem, it is fundamental to transform content-oriented MFE into a user-oriented one, that is, mining preference-related information and filtering out preference-independent noisy data. In practice, semantic-rich entities in multimodal content are highly correlated to the user purchase behavior (*cf*. Figure 1). Therefore, extracting these semantic entities rather than the full multimodal deep features facilitates the transformation of MFE. Furthermore, the user sentiment information contained in item reviews is a crucial and typical preference-related feature of textual modality, as it reflects the overall quality of the item, and the users always tend to purchase high-quality items. Consequently, capturing sentiment information can further transform the MFE module into user-oriented one.

After achieving the transformation of MFE, the next step is to establish an association between MFE and UIM. Methodologically, EF is a reasonable option as it uses interaction data to jointly optimize MFE and UIM. However, in fact, multimedia recommendations are generally applied in sparse and cold-start scenarios, which means that the large number of learnable parameters in the MFE module are difficult to be optimized. From this view, SF is considered as the overall framework, and the MFE is treated as a pre-processing module. Moreover, multimodal semantic correlation, which seamlessly associates the MFE and UIM through interaction relationship, is proposed. Figure 2 presents a simple example to show the multimodal semantic correlation and its importance for modeling user preferences. The left subfigure shows that the interaction sparsity makes the measurement of the similarity between nodes difficult (as there is no path between u_1 and u_2). In the right subfigure, after incorporating the semantic entities, the similarity between u_1 and u_2 can be accurately measured due to the fact that a path $<u_1,i_2,e_2,i_3,u_2>$ emerges between them (the similarity is referred to as the multimodal semantic correlation $C_{u_1u_2}$ between u_1 and u_2). Similarly, $C_{i_1i_2}$ represents the semantic correlation between i_1 and i_2. In addition, semantic correlation

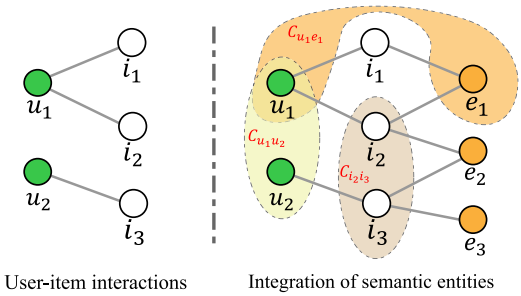

Fig. 2. Illustration of multimodal semantic correlation, where u, i, and e denote the user, item, and semantic entity, respectively, and C_{ab} denotes the semantic correlation between a and b.

can also represent the user preference for entities (*e.g.*, $C_{u_1e_1}$). e_1 is clearly the best match for u_1's preference, as the most paths between them exist.

In order to better quantify the aforementioned semantic correlation, the **Graph Convolution Network (GCN)** [17] is the optimal choice as the recent study [9, 39] has demonstrated its outperformance in capturing high-order correlation between nodes on graph. In addition, the modeling of this correlation can be further enhanced by fine-grained weighting of neighbor aggregation in GCN, which is conventionally implemented by constructing attention networks [33]. However, in the multimodal recommendation scenarios, the weighting strategy of self-attention is suboptimal, because it ignores the capture of preference-related sentiment information (or overall quality) hidden in the items (evidence in Section 4.3.5). From this view, it is a better option to mine sentiment information from item reviews and then use it to weight neighbor aggregation. Moreover, the weights obtained by sentiment information are static parameters, and therefore they do not increase the training difficulty and computational burden of the model.

Generally speaking, in this paper, we propose a GCN-based multimodal recommender method, referred to as MEGCF. First, we employ advanced deep learning techniques to mine semantic entities from multimodal content and seamlessly integrate them into the user-item interaction graph. Next, we construct a symmetric linear GCN module to perform high-order message propagation on the graph, thus modeling multimodal semantic correlation among nodes and extending it to higher-order. Finally, in order to make full use of preference-related features in textual modality, we utilize sentiment analysis techniques to extract sentiment features from item reviews and propose a sentiment weighting strategy to enhance the graph convolution operations. We conduct extensive experiments on three public datasets, and the results show that our proposed method significantly outperforms the state-of-the-art multimodal recommender method (GRCN [42]) and GCN-based CF method (SGL[45]). Furthermore, we validate the effectiveness of each component in MEGCF through sufficient ablation experiments.

We summarize the contributions of this paper as follows:

- We highlight the mismatch problem between multimodal feature extraction and user interest modeling in existing multimodal recommender methods, which contaminates embeddings and makes the models non-robust. To solve this problem, we propose to model multimodal semantic correlation and extract sentiment information.
- We propose a novel GCN-based multimodal recommender method, referred to as MEGCF, which utilizes multimodal semantic entity extraction and sentiment-weighted symmetric linear GCN module to achieve simultaneous capture of high-order multimodal semantic correlation and CF signals.

- We conduct extensive experiments on three real-world datasets to validate the state-of-the-art performance of MEGCF. In addition, further ablation experiments are performed to verify the effectiveness of each component in MEGCF. To facilitate subsequent research, we release the complete code and data of MEGCF at *https://github.com/hfutmars/MEGCF*.

2 RELATED WORK

In this section, we briefly review three types of recommender methods that are most relevant to our work: traditional Collaborative Filtering (CF)-based methods, Graph Convolution Network (GCN)-based recommender methods, and multimodal recommender methods.

2.1 Traditional CF-based Recommender Methods

Since the base framework of our work is the CF approach, we present CF and related work based on it here. CF [26] assumes that similar users exhibit similar interests in items they historically interacted with. **Matrix Factorization (MF)** [50] is the pioneering work of CF, which generates dense embedding representations for users and items by mapping their ID information, and reconstructs the unobserved interactions between users and items with the inner product of their embeddings. BiasSVD [12] assumes that MF fails to accurately capture the difference in preferences between users (or items) and introduces bias terms on top of MF to compensate for the weakness of embedding expressiveness. However, the model performance of MF and BiasSVD strongly depends on sufficient interaction data, that is, they suffer from the problem of sparsity and cold-start. To tackle this problem, one promising solution is to leverage side information that may preserve clues related to user preferences. SVD++ [18] and FISM [13] incorporate collections of items that users have historically interacted with into user embeddings, and their effectiveness in explicitly modeling interactions have been validated in subsequent work [9, 39]. Unlike merely using ID information, another family of CF methods focuses on mining user preferences from interaction-independent information. For example, SVDFeature [3] is a feature-based MF model that incorporates user and item attributes into the embeddings to enhance their expressiveness. The knowledge graph-based recommender methods [34, 35] extract item-related attribute entities and relations from external knowledge graphs into the item embeddings, thus achieving the association between user preferences and knowledge. In contrast to the above CF methods centered on enhancing embedding, NCF [11] and DICF [49] use deep neural networks instead of simple inner products, enhancing the modeling of complex interactions.

2.2 GCN-based Recommender Methods

In Section 3.2, we propose an improved GCN module for extending low-order features to higher-order. Thus, we present GCN-based recommender methods here. GCN [7, 17] is a deep neural network proposed for graph-structured data. In recent years, it has been widely studied and led to satisfactory results in recommender systems. The core paradigm of GCN is to iteratively aggregate neighbor nodes into the embeddings of the target nodes, thus explicitly capturing important high-order connectivities on the graph.

To the best of our knowledge, GC-MC [1] is the first approach that applies GCN to recommender systems, which leverages graph convolution to aggregate one-hop neighbor nodes into the embeddings of the target nodes. PinSAGE [51] and NGCF [39] integrate high-order neighbor nodes into the embedding generation of the target node. Specifically, PinSAGE combines random walks and GCN to achieve embedding generation on large-scale item-item graphs, demonstrating that GCN-based methods can be efficiently applied to web-scale recommendation scenarios. NGCF [39] is a recommender framework which combines GCN and MF. It uses a layer aggregation mechanism [48] to concatenate the embeddings of all the layers as the final node representations, in order to

capture the semantic information preserved by different graph convolution layers. Some subsequent studies [2, 9, 22] deduced that the simplified GCN [44] is better suited for modeling interactions in recommendation scenarios. LR-GCCF [2] and LightGCN [9] can be broadly seen as the lightweight versions of NGCF. LRGCCF removes the nonlinear activation function from NGCF and re-analyzes the layer aggregation mechanism in NGCF from a residual perspective. LightGCN removes both the nonlinear activation function and the weight transformation matrices from NGCF. In addition, extensive experiments show that a more concise structure of LightGCN can achieve a better performance. For convenience, we refer to the GCN module in LightGCN as the linear GCN in this work. Essentially, the Graph Laplacian Norm in GCN can be considered as a scaling of node degree (or popularity) at a fixed granularity, while users have different sensitivities to popularity features. Consequently, JMPGCF [21] is proposed to construct different Graph Laplacian Norms for GCNs, in order to capture multi-grained popularity features, and thus better model user preferences on popularity. Although the previously mentioned GCN-based approaches achieve a high performance by explicitly modeling high-order connectivities, this schema of graph convolution ignores the modeling of the diversity of user intents. Therefore, DGCF [40] is proposed to construct a graph disentangling module, in order to iteratively refine the intent-aware interaction graphs and factorial representations. Some researchers have recently tried to fuse contrastive learning and GCN to implement recommendation. For instance, SGL [45] is a model-agnostic self-supervised contrastive learning framework which incorporates node self-discrimination task into recommendation module and jointly learns them. Thus, it alleviates the long-tail problem and enhances the model robustness to noisy interactions.

It is important to mention that the GCN module in the proposed MEGCF is significantly different from the above-mentioned GCN in terms of graph convolution operations and overall structure. More precisely, we incorporate sentiment weighting strategy and popularity features based on linear graph convolution. In addition, for the overall structure, we use two symmetric versions of this GCN to construct the final module for embedding generation.

2.3 Multimodal Recommender Methods

Considering that the motivation of this study is to address the mismatch problem between multimodal feature processing and user preference modeling, *i.e.*, it focuses on how to improve the utilization of multimodal data in recommendation scenarios. Thus, the existing work on multimodal recommendations [31] is presented. Multimodal recommender methods can be considered as content-enriched ones [46] that leverage multimodal content to assist in the recommendation. The overall framework of the existing multimodal recommender efforts can be roughly divided into two categories: **Separated Framework (SF)** and **End2end Framework (EF)**. SF separates the two modules, multimodal feature processing and user preference modeling, and uses the multimodal features obtained in the former module to enrich the embedding representation in the latter module. On the contrary, EF fuses these two modules and jointly learns the two tasks of multimodal feature processing and user preference modeling in order to perform complementarity between them.

First, we introduce the SF-based multimodal recommender methods. VBPR [8] is an early approach which uses **Convolution Neural Network (CNN)** pre-trained on ImageNet to extract deep visual features of images and incorporate them into feature representations of items in the MF framework. VPOI [36] incorporates pre-extracted deep visual features into the PMF [25] framework in order to achieve POI check-in recommendations. Inspired by the fact that user preferences tend to exhibit significant variability across different modalities, increasing efforts focus on simultaneously capturing clues of user preferences on multiple modalities. For instance, CKE [52] incorporates knowledge graphs, visual features, and textual features into embedding representation.

MMGCN [43] is a GCN-based multimodal recommender method, which constructs three GCN modules to model user preference on visual, textual, and audio modalities, respectively. Compared with CKE, MMGCN can mine higher-order multimodal similarity features, thus achieving better recommendation results. Based on MMGCN, MGAT [28] constructs the attention network to adaptively calculate the weights of user preferences on different modalities. MKGAT [27] leverages the multimodal contents to construct a multimodal knowledge graph. In addition, the tail nodes of the knowledge graph are dense vectors represented by fusing the deep features of different modalities. These tail nodes are referred to as multimodal entities in MKGAT, which is fundamentally different from the proposed multimodal semantic entities that do not use deep features but extract semantic-rich entities from multimodal content. A recent GCN-based multimodal recommender method, HUIGN [41], constructs a hierarchical graph structure and designs two types of information aggregation modules (*i.e.*, intra-level and inter-level aggregation) to model multi-level user intents. Therefore, it ensures the generation of high-quality user and item representations.

Afterwards, we introduce EF-based multimodal recommender methods. DVBPR [14] is an extended version of VBPR which integrates the CNN module into the MF module to jointly train image representations and recommendation modules in an end-to-end manner. ConvMF [15] is a fusion model of CNN and MF, in which the CNN module captures the contextual information of item reviews. Therefore, it improves the prediction accuracy. DeepCoNN [53] constructs two parallel CNN modules to learn the behavioral features of users and the attribute features of items from user-related and item-related reviews, respectively. MRG [30] is a multi-task learning model which models both the review generation module and the rating prediction module. It also jointly learns both modules to better mine user preferences on textual modality. Methodologically, the EF-based approach incorporates interaction data into the multimodal feature processing to guide the mining of preference-related multimodal features, which is supposed to be stronger than the SF-based approach. However, we argue that they still suffer from some limitations. Specifically, multimodal recommender methods are mostly applied in sparse and cold-start recommendation scenarios, which means that the large number of learnable parameters introduced by EF are difficult to be optimized. In addition, integrating multimodal feature processing modules on general recommendation modules degrades the efficiency of model training and inference. Therefore, in this paper, we adopt the idea of SF to implement MEGCF.

Despite the progress of these works, they all essentially use complete multimodal deep features to participate in the feature representations of items while ignoring the mismatch problem between multimodal feature processing and user preference modeling, which results in embedding contamination.

3 METHODOLOGY

In this section, we present the overall framework of the MEGCF model shown in Figure 3, which can be divided into three main components: (1) the multimodal semantic entity extraction layer for extracting semantic-rich entities from multimodal data, and then incorporating them into the user-item interaction graph; (2) the sentiment-weighted symmetric linear Graph Convolution Network (GCN) module for capturing both high-order multimodal semantic correlation and CF signal during embedding generation; and (3) the model prediction&optimization layer for estimating the preference scores of user-item pairs and updating model parameters.

3.1 Multimodal Semantic Entity Extraction Layer

In order to reduce the negative impact of redundant information in multimodal data with low relevance to user preferences, we propose to extract semantic entities from multimodal data and associate them with items, as semantic entities have the potential to be more interest-provoking

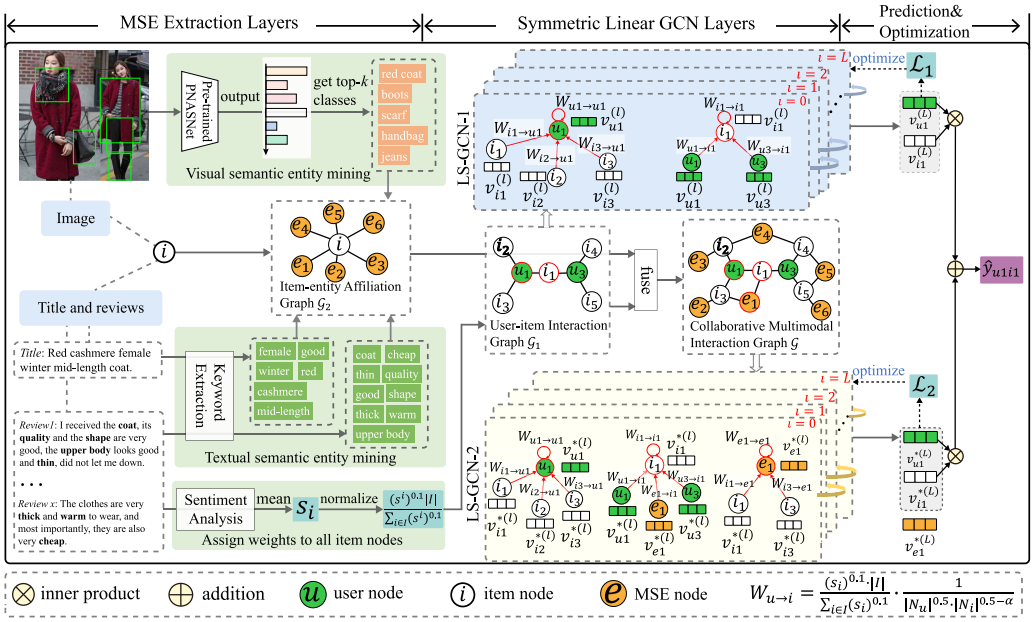

Fig. 3. Illustration of the proposed MEGCF. The target user and item are u_1 and i_1, MSE denotes multimodal semantic entity, and L is the max number of graph convolution layers.

for users than other modal information such as background, position, angle, brightness, *etc.* In the following, we detail the specific extraction of semantic entities for visual and textual modalities, respectively, and utilize them and the user-item interaction graph to construct a collaborative multimodal interaction graph.

3.1.1 Visual Semantic Entities Extraction. In order to ensure efficiency in entity extraction, we use the technique of image classification rather than object detection to extract visual semantic entities from images. Specifically, for an item i, we feed its corresponding image into a PNASNet model [20] (an advanced image classification method) pre-trained on the ImageNet dataset. The model then outputs a probability distribution over 1,000 categories, and we take the top-ranked categories as the semantic entities in the image. These entities can also be understood as the most likely sub-objects present in the image. Finally, we perform the above operations on all the items to obtain $\mathcal{E}_V$, which is the set of semantic entities on the visual modality.

3.1.2 Textual Semantic Entities Extraction. The textual data of an item i consist of a title and some reviews, on which we first perform pre-processing operations, including special characters removal, words segmentation, and stop words removal. Afterwards, as the title is informative and objective, we directly use the pre-processed words as textual entities in the title. Reviews express users' experiences and sentiment about the item i, and contain relatively less information. In addition, they are more subjective than titles. Therefore, we use the SGRank model [4] (an advanced keyword extraction method) to further extract keywords from the pre-processed reviews, and its outputs are the semantic entities in the reviews. Finally, we iteratively perform the above operations on all the items in order to obtain the set of semantic entities on the text modality, $\mathcal{E}_T$.

Compared with the direct utilization of full multimodal deep features, semantic entity extraction can better reduce the preference-independent multimodal content. However, this approach

still has limitations in entity detection accuracy. More precisely, when the neural network model pre-trained in the upstream task is directly migrated to the recommendation task, there is a significant degradation in entity detection accuracy (wrong or missing detection) occurs. Empirically, fine-tuning the pre-trained model using partially labeled data from the recommendation task is methodologically feasible. However, the multimodal data in the recommendation dataset are unlabeled. Despite the problem of wrong or missing detection, the proposed MEGCF is significantly stronger than most of the multimedia recommender methods (cf. Table 2). In addition, there is a high probability that MEGCF will be further enhanced if we can improve the accuracy of entity extraction. We leave it for future considerations.

3.1.3 Collaborative Multimodal Interaction Graph. First, user-item interactions can be translated into a bipartite graph structure $\mathcal{G}_1 = \{(u, r_{ui}, i)|u \in \mathcal{U}, i \in \mathcal{I}\}$, where $\mathcal{U}$ and $\mathcal{I}$ are the user and item sets, respectively, $r_{ui} = 1$ denotes that there is an interaction between user u and item i, otherwise $r_{ui} = 0$. Then, based on the affiliation between item $\mathcal{I}$ and the multimodal entities $\mathcal{E} = \mathcal{E}_V \cup \mathcal{E}_T$, we construct an item-entity bipartite graph $\mathcal{G}_2 = \{(i, r_{ie}, e)|i \in \mathcal{I}, e \in \mathcal{E}\}$, where $r_{ie} = 1$ denotes that the entity e is extracted from the multimodal data of item i, otherwise $r_{ie} = 0$. Finally, as shown in Figure 3, the item nodes are used as a bridge to fuse these two bipartite graphs into a new user-item-entity tripartite graph, $\mathcal{G} = \{(u, r_{ui}, i), (i, r_{ie}, e)|u \in \mathcal{U}, i \in \mathcal{I}, e \in \mathcal{E}\}$, named collaborative multimodal interaction graph.

3.2 Sentiment-weighted Symmetric Liner GCN Layer

We design a sentiment-weighted symmetric linear graph convolution block to perform message propagation on the user-item graph $\mathcal{G}_1$ and the collaborative multimodal interaction graph $\mathcal{G}$, in order to capture the high-order CF signal and multimodal semantic correlation, respectively. In the following, we detail the process of embedding generation in this block.

3.2.1 Embedding Initialization. Following the ID embedding-based recommender models, we initialize all the users, items, and semantic entities by mapping their IDs to the corresponding dense low-dimensional vector representations as follows:

$$\mathcal{V} = \left\{v_{u_1}^{(0)}, \ldots, v_{u_{|\mathcal{U}|}}^{(0)}, v_{i_1}^{(0)}, \ldots, v_{i_{|\mathcal{I}|}}^{(0)}, v_{e_1}^{(0)}, \ldots, v_{e_{|\mathcal{E}|}}^{(0)}\right\}, \tag{1}$$

where $\mathcal{V} \in \mathbb{R}^{(|\mathcal{U}|+|\mathcal{I}|+|\mathcal{E}|)\times d}$ is the embedding matrix of all the nodes in $\mathcal{G}$, $|\mathcal{U}|$, $|\mathcal{I}|$, and $|\mathcal{E}|$ are the number of users, items, and semantic entities, respectively, d is the dimension length of embeddings. It is important to mention that the user and item nodes in $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{G}$ are parameter shared.

3.2.2 Review-based Sentiment Extraction. As the reviews are highly subjective and can reflect the users' sentiments towards the target item, we propose to use an advanced sentiment analysis technique [29] to extract sentiment information from reviews, which allows fine-grained weighting to items. We formulate the calculation of sentiment score for item i as follows:

$$s_i = \frac{\sum_{t \in T_i} f(t)}{|T_i|}, \tag{2}$$

where T_i is the review set for item i, $|T_i|$ denotes the size of T_i, $f(\cdot)$ represents the pre-trained SENTA model[29], which outputs the sentiment score from the input review, and s_i is the mean sentiment score for item i.

In subsequent message propagation, we use s_i to weight all the item nodes in $\mathcal{G}_1$ and $\mathcal{G}$.

3.2.3 Sentiment-weighted Embedding Generation. The average sentiment in an item's review directly reflects the overall quality of the item. Most users usually buy high-quality items, *i.e.*, there is a higher affinity between user node and high-quality item node. Consequently, we propose to use the average sentiment score to distinguish the importance of different neighbors during the graph convolution process, so that the overall quality information mined from item reviews can be incorporated into the node representation. Note that the Graph Laplacian Norm in GCNs essentially uses node degree (number of interactions or popularity) to assign weights for neighbor aggregations, which is significantly different from the review-based sentiment weights. Therefore, using both the sentiment weights and Graph Laplacian Norm is a better option. It is worth mentioning that GAT [33], which is a variant of GCN, uses a self-attention mechanism to weight neighbor aggregations. We argue that the weighting strategy of GAT is suboptimal under the multimodal recommendation scenarios, because it only uses interactions and ignores the capture of sentiment information (or overall quality) inherent in the items (evidence in Section 4.3.5). In addition, Linear GCNs can better capture high-order CF signals, compared with the traditional nonlinear GCNs (evidence in [9, 22]).

Based on these considerations, we devise a linear sentiment-weighted GCN structure to perform message propagation on the user-item interaction graph $\mathcal{G}_1$, in order to better capture high-order CF signals. For convenience, we refer to this structure as **LS-GCN-1**. For the target user $u1$ and item $i1$, we formulate their embedding generation in LS-GCN-1 as follows:

$$v_{i1}^{(l)} = \sum_{u \in N_{i1} \cup i1} \frac{(s_{i1})^{\gamma} |\mathcal{I}|}{\sum_{i \in \mathcal{I}} (s_i)^{\gamma}} \cdot \frac{1}{|N_{i1}|^{0.5} |N_u|^{0.5-\alpha}} \cdot v_u^{(l-1)}, \tag{3}$$

$$v_{u1}^{(l)} = \sum_{i \in N_{u1} \cup u1} \frac{(s_i)^{\gamma} |\mathcal{I}|}{\sum_{i \in \mathcal{I}} (s_i)^{\gamma}} \cdot \frac{1}{|N_{u1}|^{0.5} |N_i|^{0.5-\alpha}} \cdot v_i^{(l-1)}, \tag{4}$$

where l denotes the number of current graph convolution layers, N_u and N_i are the neighbor nodes of user node u and item node i in $\mathcal{G}_1$, respectively, $|N_u|$ and $|N_i|$ denotes the size of N_u and N_i, respectively, $\frac{(s_i)^{\gamma} |\mathcal{I}|}{\sum_{i \in \mathcal{I}} (s_i)^{\gamma}}$ is the weight allocated to item i using the sentiment score s_i in Equation (2), which can also be considered as a normalization on the sentiment score s_i, $|\mathcal{I}|$ denotes the number of items, and γ is used to smooth the sentiment score s_i (we set $\gamma = 0.1$ in experiments for simplicity), note that we set $s_i = 1.0$ when $i = u1$ in Equation (4). In addition, following the recent graph learning based method [21], we fine-tune the classical Graph Laplacian Norm $\frac{1}{|N_i|^{0.5} |N_u|^{0.5}}$ to the form of $\frac{1}{|N_i|^{0.5} |N_u|^{0.5-\alpha}}$, where $\alpha \in (0, 0.5)$ is a hyper-parameter used to control the model sensitivity to popularity information. For convenience, we term this improved Graph Laplacian Norm as popularity-aware norm (short for PN, Section 4.3.2 validates the effectiveness of PN). Specifically, when α is larger, the embedding value obtained from the graph convolution is larger, which corresponds to the fact that the model is more sensitive to the popularity.

By multimodal semantic entity extraction and the construction of multimodal collaborative interaction graphs, we can model important semantic correlation, and thus reduce the negative impact of preference-independent multimodal information. Now we move forward to capture this correlation and extend it to higher-order. We propose to construct a linear sentiment-weighted GCN structure similar to LS-GCN-1, referred to as **LS-GCN-2**. More precisely, we use LS-GCN-2 to iteratively perform message propagation over the collaborative multimodal interaction graph $\mathcal{G}$, in order to bridge multimodal entity information to user representations using item nodes, while user-item interactions can, in turn, facilitate the learning of multimodal semantic correlation.

We first formulate the embedding output at l-th layer in LS-GCN-2 for the target user node $u1$ as follows:

$$v^{*(l)}_{u1} = \sum_{i \in N_{u1} \cup u1} \frac{(s_i)^\gamma |\mathcal{I}|}{\sum_{i \in \mathcal{I}} (s_i)^\gamma} \cdot \frac{1}{|N_{u1}|^{0.5} |N_i|^{0.5-\alpha}} \cdot v^{*(l-1)}_i, \tag{5}$$

where $v^{*(0)}_i$ is the initialized embedding of i in Equation (1), which is equivalent to $v^{(0)}_i$, note that $v^{*(l)}_i$ is inherently different from $v^{(l)}_i$ in Equation (4) because $v^{*(l)}_i$ is incorporated with the message from multimodal semantic entities.

We then present the embedding generation for the target entity $e1$ as follows:

$$v^{*(l)}_{e1} = \sum_{i \in N_{e1} \cup e1} \frac{(s_i)^\gamma |\mathcal{I}|}{\sum_{i \in \mathcal{I}} (s_i)^\gamma} \cdot \frac{1}{|N_{e1}|^{0.5} |N_i|^{0.5-\alpha}} \cdot v^{*(l-1)}_i, \tag{6}$$

Note that we set $s_i = 1.0$ when $i = e1$.

Since in $\mathcal{G}$, the neighbors of the item nodes include both user nodes and entity nodes, separating message aggregation for the neighbor nodes based on the node type is necessary to generate the item embedding representations. We finally formulate the embedding generation in LS-GCN-2 for the target item $i1$ as follows:

$$v^{*(l)}_{i1} = \sum_{u \in N^{(u)}_{i1} \cup i1} \frac{(s_{i1})^\gamma |\mathcal{I}|}{\sum_{i \in \mathcal{I}} (s_i)^\gamma} \cdot \frac{v^{*(l-1)}_u}{|N_{i1}|^{0.5} |N_u|^{0.5-\alpha}} + \sum_{e \in N^{(e)}_{i1} \cup i1} \frac{(s_{i1})^\gamma |\mathcal{I}|}{\sum_{i \in \mathcal{I}} (s_i)^\gamma} \cdot \frac{v^{*(l-1)}_e}{|N_{i1}|^{0.5} |N_e|^{0.5-\alpha}}, \tag{7}$$

where $N^{(u)}_{i1}$ and $N^{(e)}_{i1}$ denote the user and entity neighbor nodes of item $i1$ in $\mathcal{G}$, respectively, and $v^{*(0)}_e$ is the initialized embedding of the entity e in Equation (1), which is equivalent to $v^{(0)}_e$.

3.3 Model Prediction & Optimization Layer

3.3.1 Prediction Function. In GCN, the node embeddings obtained at layer l already preserve the information from all the previous layers. Therefore, we choose the outputs of the last layer as the final representations of all the nodes. Considering an L-layer GCN here, for a target user u and item i, we apply the inner product to calculate the user's preference score for the item as follows:

$$\hat{y}_{ui} = (v^{(L)}_u)^T \cdot v^{(L)}_i + (v^{*(L)}_u)^T \cdot v^{*(L)}_i. \tag{8}$$

3.3.2 Objective Function. In order to optimize the MEGCF, we select the BPR loss [24] as a base objective function, which is used in a wide range of recommendation methods. The core idea of BPR loss is that the preference score between the observed user-item pair is higher than that of the unobserved one.

Firstly, in order to ensure the learning of high-order CF signal, we construct the BPR loss using the embeddings output from LS-GCN-1, that is, the embeddings in Equations (3) and (4).

$$\mathcal{L}_1 = \sum_{(u,i,j) \in O} -ln\sigma([v^{(L)}_u]^T \cdot v^{(L)}_i - [v^{(L)}_u]^T \cdot v^{(L)}_j) + \lambda_1 \cdot ||\mathcal{H}_1||^2_2, \tag{9}$$

where $O = \{(u,i,j)|(u,i) \in \mathcal{R}^+, (u,j) \notin \mathcal{R}^+\}$ is the full training data, $\mathcal{R}^+$ denotes the full observed user-item interactions in $\mathcal{G}_1$, $\sigma(\cdot)$ is a sigmoid function, $\mathcal{H}_1 = \{\mathcal{V}_u{}^{(L)}, \mathcal{V}_i{}^{(L)}\}$ denotes the trainable parameters in this step, $\mathcal{V}_u{}^{(L)} = \{v^{(L)}_{u_1}, \ldots, v^{(L)}_{u_{|\mathcal{U}|}}\}$ and $\mathcal{V}_i{}^{(L)} = \{v^{(L)}_{i_1}, \ldots, v^{(L)}_{i_{|\mathcal{I}|}}\}$ are the user and item embeddings obtained at L-th layer in LS-GCN-1, respectively, and λ_1 is the coefficient of L_2 regularization for $\mathcal{H}_1$.

Afterwards, in order to facilitate the capture of high-order multimodal semantic correlations, we utilize the embeddings output from LS-GCN-2 (*i.e.*, the embeddings in Equations (5) and (7))

to construct the corresponding BPR loss:

$$\mathcal{L}_2 = \sum_{(u,i,j)\in O} -ln\sigma([v^{*(L)}_u]^T \cdot v^{*(L)}_i - [v^{*(L)}_u]^T \cdot v^{*(L)}_j) + \lambda_2 \cdot ||\mathcal{H}_2||_2^2, \tag{10}$$

where $\mathcal{H}_2 = \{\mathcal{V}^{*(L)}_u, \mathcal{V}^{*(L)}_i\}$ denotes the trainable parameters in this step, $\mathcal{V}^{*}{}_u{}^{(L)} = \{v^{*(L)}_{u_1}, \ldots, v^{*(L)}_{u_{|\mathcal{U}|}}\}$ and $\mathcal{V}^{*}{}_i{}^{(L)} = \{v^{*(L)}_{i_1}, \ldots, v^{*(L)}_{i_{|\mathcal{I}|}}\}$ are the user and item embeddings obtained at L-th layer in LS-GCN-2, respectively, and λ_2 is the coefficient of L_2 regularization for $\mathcal{H}_2$. It is worth mentioning that since the embeddings of users and items are incorporated with the information of multimodal entities through multiple message propagation, the optimization of this objective function can be considered as using the interactions to support the learning of user preferences over multimodal semantics.

Finally, we propose an objective function to jointly learn the Equations (9) and (10) as follows:

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2. \tag{11}$$

We adopt the mini-batch Adam optimizer [16] to minimize the loss in Equation (11) and update the model parameters.

3.4 Complexity Analysis of MEGCF

Here we analyze the complexity of MEGCF. To the best of our knowledge, LightGCN is the most efficient GCN-based recommendation model. Therefore, we compare its complexity with that of MEGCF. As for the model size, the model parameters introduced by LightGCN are the initialized embeddings of users and items, while MEGCF additionally introduces the embeddings of entity nodes. In the following, we analyze the time complexity of MEGCF and LightGCN for the complete model training process.

Assuming that the number of edges on the user-item interaction graph $\mathcal{G}_1$ and the multimodal interaction graph $\mathcal{G}$ are $|\mathcal{E}|$ and $|\mathcal{E}_m|$, respectively, d denotes the embedding length, s represents the number of training epochs, B is the size of each training batch, and L denotes the depth of GCN layers. Their computational complexity comes mainly from two parts: (1) the graph convolution process, and (2) the calculation of BPR loss.

- For the graph convolution process, the time complexities of MEGCF to perform one graph convolution on $\mathcal{G}_1$ and $\mathcal{G}$ are $O(2|\mathcal{E}|)$ and $O(2|\mathcal{E}_m|)$, respectively. Thus, its complexity in the whole training process is $O(2(|\mathcal{E}| + |\mathcal{E}_m|)Lds\frac{|\mathcal{E}|}{B})$, while the complexity of LightGCN is $O(2|\mathcal{E}|Lds\frac{|\mathcal{E}|}{B})$.
- For the calculation of BPR loss, the scoring prediction is the core operation to be considered. Note that both MEGCF and LightGCN use a simple inner product as the prediction function and its complexity is $O(d)$. Thus, the time complexity for LightGCN at this part in the whole training process is $O(2|\mathcal{E}|ds)$. Since MEGCF needs to perform scoring predictions on $\mathcal{G}_1$ and $\mathcal{G}$, respectively ($cf.$ Equations (9) and (10)), the corresponding time complexity of MEGCF is twice that of LightGCN, $i.e.$, $O(4|\mathcal{E}|ds)$.

Therefore, the overall training complexity of the proposed MEGCF is close to $O(2(|\mathcal{E}| + |\mathcal{E}_m|)Lds\frac{|\mathcal{E}|}{B} + 4|\mathcal{E}|ds)$, while the complexity of LightGCN is $O(2|\mathcal{E}|Lds\frac{|\mathcal{E}|}{B} + 2|\mathcal{E}|ds)$.

4 EXPERIMENT

In this section, we conduct experiments to evaluate our proposed MEGCF, and some ablation studies to verify the effectiveness of each component in MEGCF. We aim to answer three main research questions as follows:

Table 1. Statistics of the Datasets, where # VE and # TE Denote the Number of Visual and Textual Entities Introduced in Section 3.1, Respectively, and Density is Calculated by Using $\#Interaction/(\#User \times \#Item)$

Dataset	# User	# Item	# Interaction	Density	# VE	# TE
Beauty	15,576	8,678	139,318	0.00103	1,080	11,450
Art	25,165	9,324	201,427	0.00086	962	11,215
Taobao	12,539	8,735	83,648	0.00076	1,127	8,476

- **RQ1:** How does our proposed MEGCF perform compared with the state-of-the-art baselines?
- **RQ2:** Whether the components (modality-specific semantic correlation, symmetric linear GCN structure, sentiment-weighted neighbor aggregation, and joint loss function) in MEGCF are effective?
- **RQ3:** Whether the capture of multimodal semantic correlations is helpful for modeling modality-level item similarity and user preference?

4.1 Experimental Settings

4.1.1 Datasets. Since our work aims to study multimodal data processing in recommendations, we choose two real-world datasets from *Amazon.com* introduced by [23], **Beauty** and **Arts_crafts_and_Sewing** (short for **Art**)[2]; both of them contain images, titles, and reviews. Besides, we select another fashion collocation dataset, **Taobao**,[3] which is published in the Tianchi competition. This dataset contains images and titles, except reviews, so the MEGCF run on this dataset is not equipped with the strategy of sentiment-weighted neighbor aggregation. To ensure the quality of these three datasets, we apply the 5-core setting, that is, retaining that all users and items have at least five interactions.

We present the details of these three datasets in Table 1. Following the convention setting [8, 11, 49], we apply the *leave-one-out* evaluation [24] to randomly sample one item for each user to form the test set and another one to form the validation set, and the remaining interaction data to serve as the training set.

4.1.2 Evaluate Metrics. We select two protocols: *Hit Ratio (HR)* and *Normalized Discounted Cumulative Gain (NDCG)*, which are widely used in recent works [10, 11, 49] to evaluate model performance. Specifically, we compute the average $HR@k$ and $NDCG@k$ for each user in the test set. Note that for each user, we randomly sample 99 items from all items that the user has not interacted with as negative samples.

4.1.3 Baselines. To demonstrate the effectiveness of our proposed MEGCF, we compare it with the following baselines:

- **BPRMF** [50]: **Matrix Factorization (MF)** is a classical collaborative filtering method, which is widely used as a recommender baseline. BPRMF optimizes MF using BPR loss.
- **SVD++** [18]: this is a variant of MF, which integrates the historical interactions into user embeddings. It can also be viewed as a one-layer linear GCN that only passes messages for user nodes. To ensure fairness, we employ BPR loss to optimize this baseline.

[2] http://deepyeti.ucsd.edu/jianmo/amazon/index.html.
[3] https://tianchi.aliyun.com/competition/entrance/231506/information.

- **VBPR** [8]: this model incorporates visual features into the item representations and applies the MF framework to predict the preference scores of user-item pairs.
- **CKE** [52]: This model incorporates visual features, textual features, and knowledge graph into the item representations, and uses MF as the overall framework. In the experiments, we consider only visual and textual features since there is no available knowledge graph in the datasets.
- **NGCF** [39]: this model adopts a nonlinear GCN to iteratively perform message propagation on the user-item graph and concatenates the embedding obtained from each GCN layer as the final representations of the user and item nodes.
- **MMGCN** [43]: this is a multimodal recommendation model, which considers features of visual, textual, and audio modalities. It also applies three nonlinear GCNs to perform message passing on the user-item graphs that hold data of different modalities, respectively, so as to learn fine-grained modality-specific user preferences. It finally fuses embeddings of different modalities as the final representations of users and items. In the experiment, we consider only visual and textual features due to the limitation of the datasets.
- **LightGCN** [9]: this is the state-of-the-art GCN-based CF model, which incorporates a linear GCN into CF scenarios and uses the summation of the embeddings obtained at each layer as the final representation.
- **GRCN** [42]: this is the state-of-the-art multimodal recommendation method, whose main framework can be viewed as a linear GCN, where multimodal features of items are used to weight the neighbor aggregation, and finally the output of each graph convolution layer is summed and concatenated with the multimodal features as the final node representations.

4.1.4 Hyper-parameter Settings. For all methods of comparison, we set the embedding size and batch size to 64 and 2048, respectively. We tune the learning rate in $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ and search the coefficient of L_2 regularization in $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. For GCN-based methods, *i.e.,* NGCF, MMGCN, LightGCN, GRCN, and our proposed MEGCF, we tune the number of graph convolution layers in {1, 2, 3, 4, 5, 6}. Besides, we use the Xavier initializer [5] to achieve the embedding initialization for all models.

4.2 Overall Comparison (RQ1)

To evaluate our proposed MEGCF, we compare it with traditional CF models (BPRMF and SVD++), GCN-based CF models (NGCF and LightGCN), and multimodal-based models (VBPR, CKE, MMGCN, and GRCN). Table 2 and Figure 4 report the performance of all the models. We obtain the following findings:

- BPRMF performs the weakest in all the cases, which indicates that the use of simple inner product strongly depends on sufficient interaction and makes it difficult to model complex interaction connectivity in sparse scenarios. SVD++ outperforms BPRMF on the three datasets, which demonstrates that explicitly incorporating historical interactions into the user embedding is helpful for modeling user preferences.
- The GCN-based methods (NGCF, MMGCN, LightGCN, GRCN) consistently outperform BPRMF and SVD++, which demonstrates the effectiveness of explicitly capturing high-order CF signals. In addition, LightGCN achieves a significant improvement over NGCF on the three datasets, which is due to the fact that the linear GCN used by LightGCN is more suitable for capturing high-order CF signals than the nonlinear GCN used by NGCF (evidence is also given in [9]).
- The multimodal baselines (VBPR, CKE, and MMGCN) always outperform SVD++, which demonstrates the effectiveness of modeling modality-level user preferences in solving

Table 2. Overall Performance Comparison

Metric	Models	Beauty			Art			Taobao		
		$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$
HR@k	BPRMF	0.4274	0.5173	0.6231	0.6333	0.7052	0.7829	0.3215	0.4049	0.5155
	SVD++	0.4584	0.5520	0.6659	0.6530	0.7425	0.8285	0.3374	0.4293	0.5466
	VBPR	0.4722	0.5670	0.6665	0.6699	0.7464	0.8262	0.3464	0.4364	0.5512
	CKE	0.4810	0.5894	0.6950	0.6719	0.7632	0.8461	0.3560	0.4550	0.5789
	NGCF	0.4853	0.5820	0.6810	0.6742	0.7541	0.8287	0.3575	0.4593	0.5841
	MMGCN	0.4934	0.6067	0.7166	0.6769	0.7702	0.8546	0.3649	0.4695	0.5902
	LightGCN	0.5002	0.6063	0.7178	0.6814	0.7639	0.8329	0.3848	0.4893	0.6237
	GRCN	0.5087	0.6204	0.7241	0.6905	0.7743	0.8532	0.3865	0.4996	0.6375
	MEGCF	**0.5439**	**0.6464**	**0.7448**	**0.7116**	**0.7902**	**0.8651**	**0.4045**	**0.5212**	**0.6516**
	%Improv.	6.92%	4.19%	2.86%	3.06%	2.05%	1.39%	4.65%	4.32%	2.21%
NDCG@k	BPRMF	0.3343	0.3634	0.3900	0.5597	0.5829	0.6025	0.2465	0.2733	0.3011
	SVD++	0.3592	0.3895	0.4157	0.5627	0.5916	0.6134	0.2523	0.2819	0.3114
	VBPR	0.3665	0.3973	0.4224	0.5830	0.6078	0.6280	0.2639	0.2928	0.3216
	CKE	0.3650	0.4002	0.4269	0.5739	0.6030	0.6245	0.2622	0.2941	0.3253
	NGCF	0.3776	0.4089	0.4339	0.5882	0.6141	0.6330	0.2658	0.2986	0.3301
	MMGCN	0.3714	0.4081	0.4359	0.5643	0.5945	0.6159	0.2709	0.3047	0.3351
	LightGCN	0.3807	0.4152	0.4435	0.5886	0.6153	0.6340	0.2840	0.3176	0.3515
	GRCN	0.3910	0.4272	0.4533	0.5937	0.6208	0.6407	0.2861	0.3225	0.3573
	MEGCF	**0.4257**	**0.4590**	**0.4838**	**0.6144**	**0.6398**	**0.6588**	**0.3020**	**0.3397**	**0.3726**
	%Improv.	8.87%	7.44%	6.73%	3.49%	3.06%	2.83%	5.56%	5.33%	4.28%

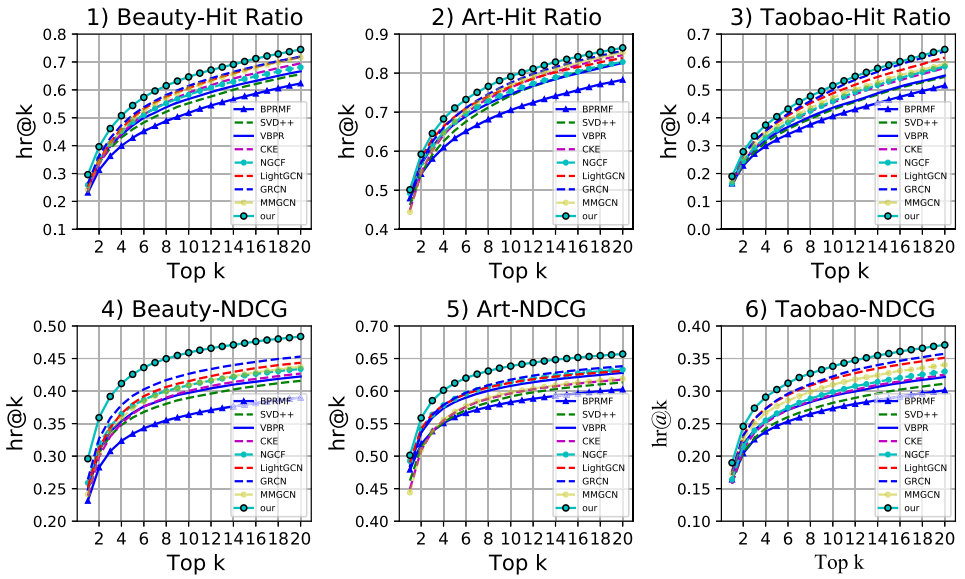

Fig. 4. Performance comparison of all the models over the different k on the three datasets.

sparsity problems. CKE slightly outperforms VBPR *w.r.t.* HR in most cases, which is owning to the additional incorporation of deep textual features in CKE. However, CKE is often weaker than VBPR *w.r.t.* NDCG especially on the Art dataset, which is attributed to the fact that the textual deep features that CKE additionally incorporates contain too much noise

unrelated to user preferences, thus contaminating the embedding representation and weakening the modeling of ranking preferences.

- MMGCN outperforms NGCF in most cases, which indicates that incorporating multimodal features while capturing high-order CF signals can further improve the model performance. Unexpectedly, in terms of NDCG on the Art dataset, MMGCN is weaker than NGCF, CKE, and VBPR. This may be due to the fact that MMGCN extends multimodal features to a higher order through the user-item interaction graph, which in turn amplifies the preference-independent information and intensifies the contamination of embeddings, thereby weakening the modeling of user preferences in terms of ranking. The results of GRCN confirm this analysis, since GRCN, combining higher-order CF signal and lower-order multimodal features, consistently achieves a high performance $w.r.t.$ NDCG.
- MEGCF achieves the optimal performance on the three datasets, which demonstrates the importance of simultaneously mining high-order multimodal semantic correlation and CF signal. In particular, MEGCF achieves maximum improvements of 8.87% and average improvements of 4.40% compared with the strongest baseline GRCN (the method with an underline). An important observation is that MEGCF achieves more remarkable improvements on NDCG than on HR, which may be attributed to the fact that the multimodal semantic correlation captured in MEGCF is more advantageous for ranking preference modeling. In-depth experiments in Section 4.3.1 further validate this analysis, as after eliminating all the multimodal information ($i.e.$, the model variant **w/o V&T**), the model performance decreases more on NDCG than on HR. It is worth mentioning that the multimodal recommender baselines (VBPR, CKE, and MMGCN) perform poorly on NDCG, while MEGCF achieves a greater improvement on NDCG. This indicates that for alleviating the embedding contamination, mining semantic correlation from multimodal content ($i.e.$, the strategy in MEGCF) is more effective than incorporating complete multimodal deep features.

4.3 Study of MEGCF (RQ2)

In this section, we aim to investigate the effectiveness of all the components in MEGCF. Specifically, we first study the impact of modality-specific semantic correlation on model performance. Then, we further decompose the symmetric GCN modules in MEGCF and study the gains they bring to the model. Next, we investigate the effectiveness of the proposed joint loss function. After that, we compare the impacts of sentiment weighting strategy on different GCN-based methods. Finally, we assess the influence of the number of graph convolution layers in MEGCF and other GCN-based baselines.

4.3.1 Is Modality-specific Semantic Correlation Helpful? MEGCF mines semantic entities from visual and textual modalities, respectively, and captures high-order multimodal semantic correlation using user-item interactions and these semantic entities. In order to study the impact of semantic correlation of different modalities on the model performance, we set up the following variants of MEGCF:

- **w/o V**: this model is obtained by removing the visual semantic entities from MEGCF.
- **w/o T**: this model is obtained by removing the textual semantic entities from MEGCF.
- **w/o V&T**: this model removes both textual and visual semantic entities, which is equivalent to the symmetric GCN module in MEGCF capturing only the high-order CF signals.

We conduct ablation experiments on the three datasets. Figure 5 and Table 3 record the model performance of the three variants and MEGCF *w.r.t.* HR@k and NDCG@k. We find that the curve of **w/o V&T** always lies at the bottom, $i.e.$, the model performs worst when the multimodal semantic

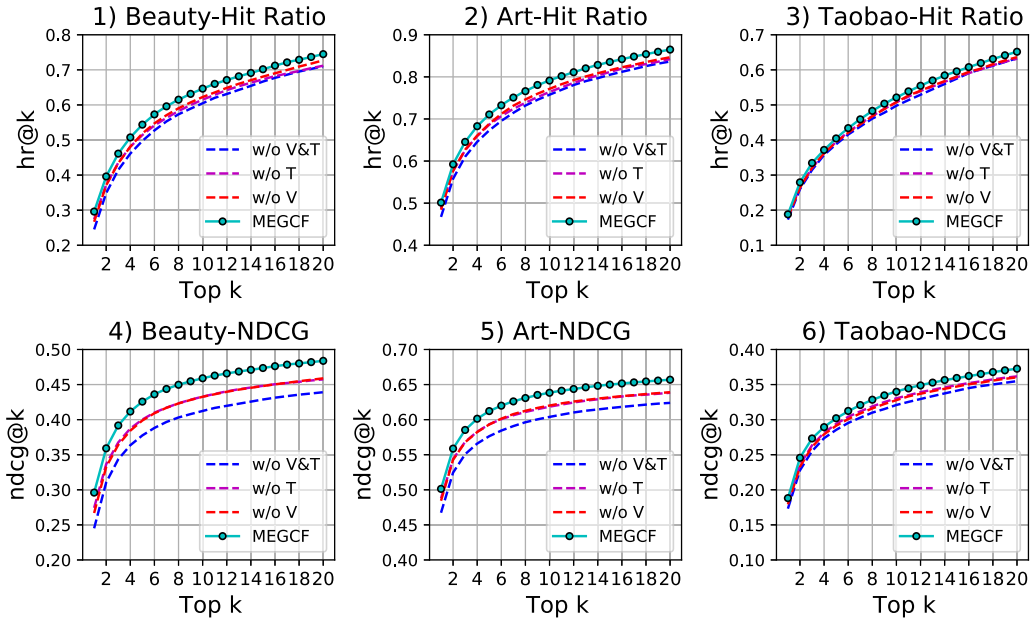

Fig. 5. Effect of modality-specific semantic correlation on MEGCF.

Table 3. Ablation Study of Components in MEGCF

Methods	Beauty				Art				Taobao			
	HR@k		NDCG@k		HR@k		NDCG@k		HR@k		NDCG@k	
	$k=10$	$k=20$	$k=10$	$k=20$	$k=10$	$k=20$	$k=10$	$k=20$	$k=10$	$k=20$	$k=10$	$k=20$
w/o V&T	0.6047	0.7105	0.4124	0.4391	0.7477	0.8250	0.6131	0.6327	0.4978	0.6329	0.3209	0.3549
w/o V	0.6226	0.7271	0.4326	0.4589	0.7716	0.8466	0.6201	0.6391	0.5076	0.6362	0.3278	0.3603
w/o T	0.6150	0.7118	0.4328	0.4574	0.7642	0.8421	0.6185	0.6383	0.5079	0.6316	0.3304	0.3617
MEGCF$_{g1}$	0.6026	0.7093	0.4145	0.4415	0.7542	0.8314	0.6145	0.6340	0.4990	0.6244	0.3241	0.3557
MEGCF$_{g2}$	0.6156	0.7345	0.4100	0.4400	0.7654	0.8448	0.6103	0.6304	0.4849	0.6217	0.3196	0.3540
w/o $\mathcal{L}_1$	0.5902	0.7171	0.3609	0.3930	0.7411	0.8455	0.5239	0.5504	0.4751	0.6355	0.2625	0.3030
w/o $\mathcal{L}_2$	0.6161	0.7214	0.4255	0.4522	0.7355	0.8139	0.5975	0.6173	0.5047	0.6387	0.3250	0.3587
w/o PN	0.6359	0.7358	0.4509	0.4763	0.7809	0.8557	0.6297	0.6493	0.5109	0.6404	0.3329	0.3665
MEGCF	0.6464	0.7448	0.4590	0.4838	0.7912	0.8648	0.6383	0.6569	0.5212	0.6516	0.3397	0.3726

entities are not used, which indicates the importance of capturing multimodal semantic correlations. Furthermore, the curves of **w/o V** and **w/o T** considering visual or textual information alone are consistently close to each other, *i.e.*, the capture of visual and textual semantic correlations provides similar gains in model performance. This may be due to the fact that in the recommendation scenarios for these three datasets (E-commerce), the visual and textual contents of the item contribute similarly to triggering the user's interaction behavior (*i.e.*, whether the user will like the item or not). MEGCF consistently outperforms **w/o V** and **w/o T**, which indicates that the visual and textual features are significantly different in triggering user interest, and thus simultaneously modeling semantic correlation on multiple modalities can better model user preferences.

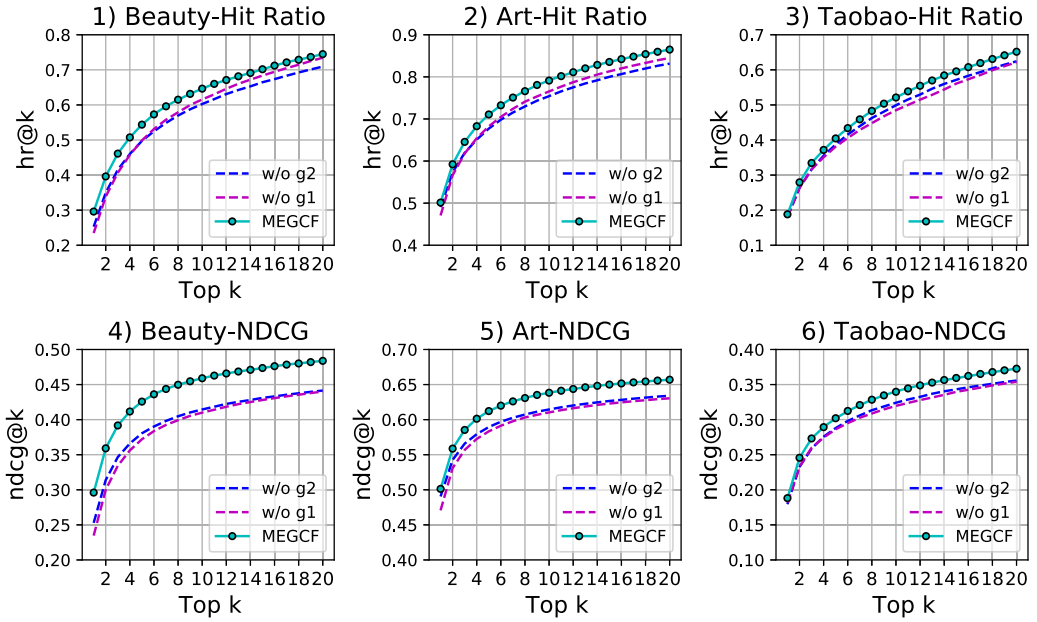

Fig. 6. Effect of symmetric linear GCN module on MEGCF.

4.3.2 Is Symmetric Linear GCN Block Helpful? In Section 3.2, we propose two linear GCN, LS-GCN-1 and LS-GCN-2 (respectively denoted by g1 and g2). They can capture high-order CF signals and high-order multimodal semantic correlations, respectively. In order to investigate the effect of them, we set up the following variants of MEGCF:

- **w/o g2**: this model retains only g1 in the symmetric linear GCN to generate the final embedding representations of the nodes.
- **w/o g1**: this model uses g2 rather than the symmetric linear GCN module for node embedding generation.
- **w/o PN**: this variant uses the classical Graph Laplacian Norm rather than the popularity-aware norm (short for PN, cf. Equation (3)).

We conduct experiments on these two variants and MEGCF using the three datasets. Figure 6 shows the top-k recommendation performance of **w/o g2**, **w/o g1**, and MEGCF, and Table 3 records the specific performance of these methods. We have the following findings:

- MEGCF consistently outperforms **w/o g2** and **w/o g1**, which indicates that there are significant differences between the CF signal and multimodal correlation captured by g1 and g2, respectively. Therefore, MEGCF combined with g1 and g2 can achieve complementarity between CF signal and multimodal correlation, and it further enhance the model performance.
- **w/o g1** is slightly weaker than **w/o g2** in most cases, which may be due to the difference between the captured CF signals in g1 and g2. More precisely, although g2 can be roughly considered as a simultaneous capture of the CF signal and semantic correlation, the CF signal in g2 is mainly used to assist in the capture of semantic correlation. Therefore, it will inevitably incorporate preference-independent multimodal noise, which makes it less pure than the CF signal captured alone in g1. This leads to the result that **w/o g1** is weaker than **w/o g2**.

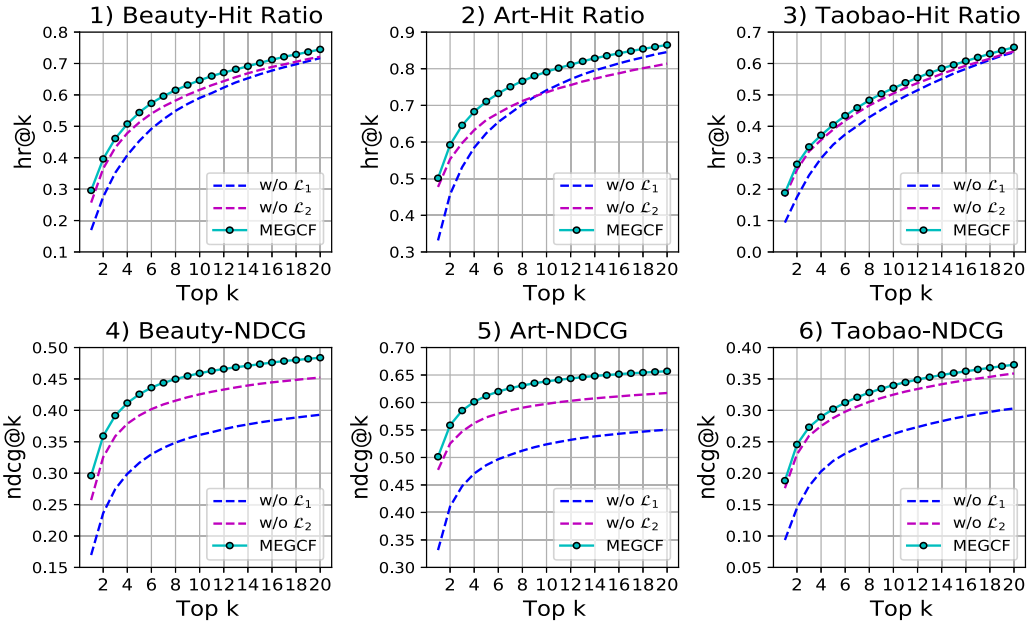

Fig. 7. Effect of joint loss function on MEGCF.

- It can be seen from Table 3 that MEGCF is stronger than **w/o PN** on the three datasets, which indicates the effectiveness of the popularity-aware norm in symmetric linear GCN module. Note that this result is consistent with that in [21].

4.3.3 Is Joint Loss Function Helpful? In Section 3.3.2, we set the corresponding loss functions $\mathcal{L}_1$ and $\mathcal{L}_2$ for optimizing the model learning for CF signals and multimodal semantic correlations, respectively, while the final joint loss function is their simple summation. In order to investigate the effectiveness of the joint loss function in MEGCF, we set up the following model variants:

- **w/o** $\mathcal{L}_1$: this variant removes $\mathcal{L}_1$ from the final loss function of MEGCF. That is, its optimization goal is to capture multimodal semantic correlations.
- **w/o** $\mathcal{L}_2$: this variant removes $\mathcal{L}_2$ from the final loss function of MEGCF, which indicates that its optimization goal is to capture high-order CF signals, while the modeling of multimodal semantic correlations is weaker.

Figure 7 and Table 3 document the trend plots and the specific values of model performance for these model variants on the three datasets, respectively. **w/o** $\mathcal{L}_2$ outperforms **w/o** $\mathcal{L}_1$ in most cases, which indicates that simply optimizing the capture of multimodal semantic correlations is insufficient. In addition, MEGCF achieves a better performance than **w/o** $\mathcal{L}_1$ and **w/o** $\mathcal{L}_2$ on the three datasets, which demonstrates the effectiveness of the strategy of using a joint loss function to simultaneously optimize both the high-order CF signals and multimodal semantic correlations.

4.3.4 Is Sentiment-weighted Neighbor Aggregation Helpful? In MEGCF, we extract the sentiment information of users from the item's reviews and then use it to assign weights to this item node, which allows the model to distinguish the importance of different neighbors during the graph convolution process. In order to investigate whether the sentiment-weighted neighbor aggregation

Table 4. Effect of the Sentiment-weighted Neighbor Aggregation on GCN-based Methods

Methods	Beauty						Art					
	HR@k			NDCG@k			HR@k			NDCG@k		
	$k=5$	$k=10$	$k=20$	$k=5$	$k=10$	$k=20$	$k=5$	$k=10$	$k=20$	$k=5$	$k=10$	$k=20$
NGCF	0.4853	0.5820	0.6810	0.3764	0.4089	0.4339	0.6742	0.7541	0.8287	0.5882	0.6141	0.6330
NGCF$_s$	0.4878	0.5828	0.6821	0.3776	0.4114	0.4354	0.6784	0.7565	0.8295	0.5911	0.6167	0.6362
%*Improv.*	0.52%	0.14%	0.16%	0.32%	0.61%	0.35%	0.63%	0.32%	0.10%	0.49%	0.42%	0.51%
MMGCN	0.4934	0.6067	0.7166	0.3714	0.4081	0.4359	0.6736	0.7681	0.8526	0.5627	0.5933	0.6148
MMGCN$_s$	0.4981	0.6092	0.7176	0.3746	0.4117	0.4379	0.6756	0.7695	0.8531	0.5685	0.5988	0.6180
%*Improv.*	0.95%	0.41%	0.14%	0.86%	0.88%	0.46%	0.30%	0.18%	0.06%	1.03%	0.93%	0.52%
LightGCN	0.5002	0.6063	0.7178	0.3807	0.4152	0.4435	0.6814	0.7639	0.8376	0.5886	0.6153	0.6340
LightGCN$_s$	0.5040	0.6100	0.7207	0.3838	0.4181	0.4460	0.6834	0.7647	0.8383	0.5897	0.6166	0.6350
%*Improv.*	0.76%	0.78%	0.14%	1.40%	1.13%	0.54%	0.29%	0.10%	0.08%	0.19%	0.21%	0.16%
w/o S	0.5332	0.6373	0.7404	0.4135	0.4472	0.4733	0.7052	0.7854	0.8587	0.6075	0.6335	0.6521
MEGCF	0.5439	0.6464	0.7448	0.4257	0.4590	0.4838	0.7116	0.7902	0.8651	0.6144	0.6398	0.6588
%*Improv.*	2.01%	1.43%	0.59%	2.95%	2.64%	2.22%	0.91%	0.61%	0.75%	1.14%	0.99%	1.03%

approach can improve the performance of MEGCF, and also to explore whether this approach is equally effective for other GCN-based recommender methods, we set up the following model variants:

- **NGCF**, **LightGCN**, and **MMGCN**: the GCN-based baselines introduced in Section 4.1.3.
- **NGCF$_s$**, **LightGCN$_s$**, and **MMGCN$_s$**: these model variants are obtained by applying the sentiment-weighted neighbor aggregation to NGCF, LightGCN, and MMGCN, respectively.
- **w/o S**: this variant is obtained by removing the sentiment weighting strategy from MEGCF.

Table 4 documents the specific performance of these methods. We have the following findings:

- NGCF$_s$, MMGCN$_s$, LightGCN$_s$, and MEGCF generally outperform NGCF, MMGCN, LightGCN, and w/o S, respectively. This demonstrates that the sentiment-weighted approach not only brings gains for MEGCF, but it is also effective for other GCN-based methods.
- The sentiment weighting strategy achieves more significant improvement on MEGCF, compared with other GCN-based methods, which may be attributed to the symmetric graph convolution structure of MEGCF. Specifically, MEGCF is equipped with two different linear GCN that both benefit from the sentiment-weighted neighbor aggregation, which results in more improvement on MEGCF.
- An important phenomenon is that the improvement of these methods on NDCG is generally higher than that on HR. In addition, the smaller the size of the recommendation list (*i.e.*, k), the higher the improvement of these methods. All these results reflect the outperformance of sentiment-weighted neighbor aggregation methods in ranking preference modeling. Specifically, these GCN-based methods with sentiment weighting will tend to rank items of interest to users more towards the top of the top-k recommendation list. Therefore, they lead to more improvement when k is smaller or when NDCG is computed.
- The sentiment-weighted neighbor aggregation method has a significantly higher improvement on the Beauty dataset (almost 1.97%) than on the Art dataset (almost 0.91%), which is coherent with the results of overall comparison in Table 2. This is probably because in the recommendation scenario corresponding to the Art dataset, using only interaction data can achieve remarkable recommendation results (HR@5 > 0.7), which means that the room for further improvement will be more limited.

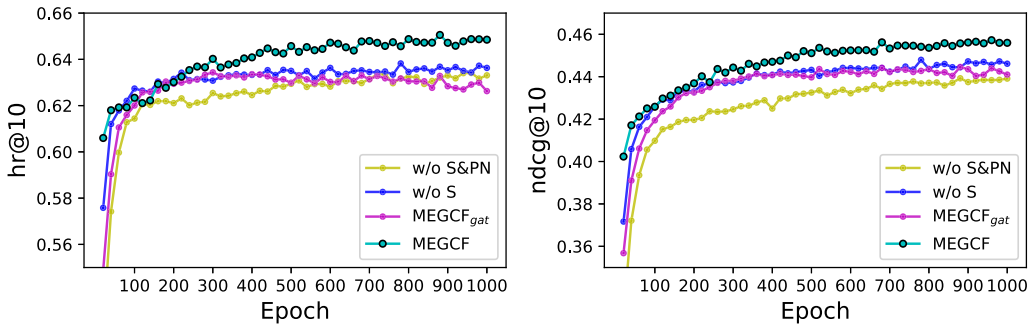

Fig. 8. Performance trends on different training epochs on the Beauty dataset.

4.3.5 Is Sentiment Weighting Superior to Self-attention Weighting? In this part, in order to compare the influences of the review-based sentiment weighting strategy and the self-attention weighting of GAT [33] on the neighbor aggregation in GCN, we set up the following model variants:

- **w/o S&PN**: this variant removes both the sentiment weighting strategy and the popularity-aware norm (short for PN) from MEGCF.
- **MEGCF**$_{gat}$: this variant uses the self-attention networks in GAT to weight the neighbor aggregation in MEGCF, rather than the sentiment weighting strategy.

Figure 8 records the performance trends of these methods on the training epochs of 0-1000. we have the following findings:

- MEGCF is significantly stronger than MEGCF$_{gat}$, which indicates that sentiment weighting outperforms self-attention weighting in GAT for user preference modeling. Furthermore, we attribute this outperformance to the utilization of both interaction data and reviews in the sentiment weighting strategy, while the weights in MEGCF$_{gat}$ are only learned from interaction data.
- The curve of MEGCF$_{gat}$ and that of **w/o S** are close. This may be due to the fact that self-attention weighting in MEGCF$_{gat}$ and **popularity-aware norm (PN)** in **w/o S** exert similar effects on the model, both amplifying the CF signal (*i.e.*, assigning higher weights to nodes with more interactions). We leave this phenomenon for future studies. In the late training period (when the training epoch is greater than 700), the performance of MEGCF$_{gat}$ starts to slightly decrease, probably because the self-attention networks introduce more learnable parameters (weight matrices), which not only increases the model complexity but also makes the model more prone to the risk of overfitting.
- MEGCF$_{gat}$ significantly outperforms **w/o S&PN** in the early stage of training, which demonstrates the effectiveness of the self-attention weighting strategy. However, in the late stage of training, **w/o S&PN** performs very close to MEGCF$_{gat}$, which may be because the light-weight GCN used in these model variants has a strong ability to mine the interaction relations, and therefore the performance gain brought by the self-attention network in MEGCF$_{gat}$ can be replaced by adequate model training in **w/o S&PN**.

4.3.6 Effect of the Number of Graph Convolution Layers. In order to better investigate the ability of MEGCF to capture multimodal correlation and CF signal, we compare the model performance of MEGCF and other GCN-based baselines (NGCF, MMGCN, and LightGCN) *w.r.t.* different graph convolution layers (we search the number of layers in {1, 2, 3, 4, 5, 6}), as shown in Figure 9. We have the following findings:

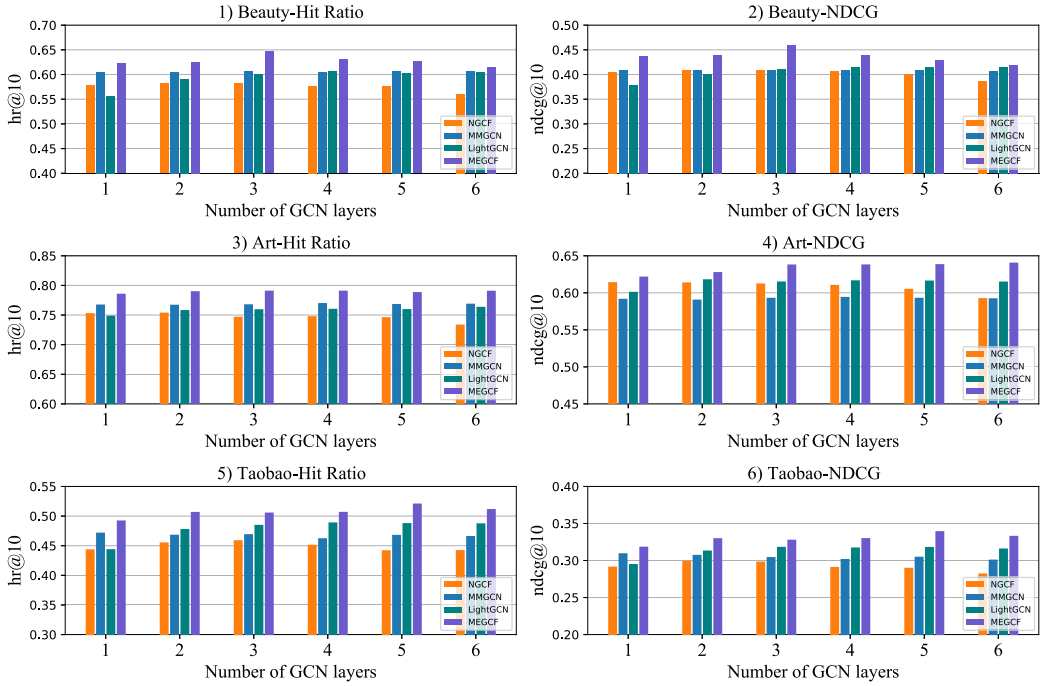

Fig. 9. Performance of GCN-based methods (NGCF, LightGCN, MMGCN, and MEGCF) *w.r.t.* the number of GCN layers on the three datasets.

- Generally speaking, MEGCF achieves the optimal performance for all layer settings on the three datasets, which further demonstrates the effectiveness of MEGCF.
- NGCF and MMGCN show relatively flat performance on the three datasets *w.r.t.* the number of layers, while LightGCN and MEGCF demonstrate a significant upward trend as the number of layers increases. We attribute such results to the fact that the nonlinear GCNs used by NGCF and MMGCN limit the capture of high-order CF signal, while LightGCN and MEGCF use linear GCN modules, which results in better performance. These results demonstrate the outperformance of the linear GCN structure for capturing high-order CF signals.
- MMGCN generally outperforms NGCF, while NDCG@10 of MMGCN is significantly weaker than that of NGCF on the Art dataset. This is attributed to the fact that the additional item multimodal deep features incorporated in MMGCN can enhance the user preference modeling. However, these features contain a considerable amount of preference-independent multimodal information, especially in the Art dataset where the negative impact of this information is greater. These results illustrate that the user preference-independent multimodal features can seriously contaminate embedding generation and ranking preference modeling.
- MEGCF consistently outperforms LightGCN significantly, which is due to the fact that MEGCF not only captures high-order CF signals but also models high-order multimodal semantic correlations. This demonstrates the importance of high-order multimodal semantic correlation in modeling user preferences. Moreover, compared with MMGCN, MEGCF, which is also based on multimodal information, exhibits a more pronounced upward trend as the number of graph convolution layers increases. This demonstrates that modeling high-order multimodal semantic correlations is more effective than existing multimodal feature processing approaches.

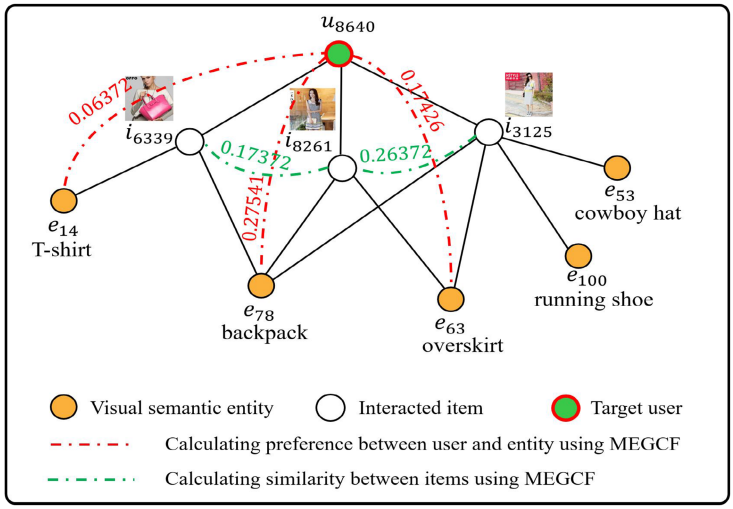

Fig. 10. Real example from Taobao dataset, the target user is u_{8640}.

4.4 Case Study (RQ3)

In order to more intuitively explain the important role of multimodal semantic correlation in modeling modality-level user preference and item similarity, we design a simple case study on the Taobao dataset as shown in Figure 10. For simplicity, we only consider semantic entity mining of visual modality. Firstly, we randomly select a user u_{8640} having an ID of 8640. Her historical interacted items are i_{6339}, i_{8261}, and i_{3125}, and we give the images corresponding to these items in Figure 10. Then, we leverage the method introduced in Section 3.1 to mine the visual semantic entities in the images. Afterwards, using all this information, a partial collaborative multimodal interaction graph with user u_{8640} as the central node can be constructed. Finally, we use the MEGCF that has been trained on the Taobao dataset to predict user preference scores for different modality semantic entities, and calculate the similarities between different items using the inner product of the embeddings. We have the following findings:

- For the semantic entities (e_{14}: T-shirt, e_{78}: backpack, and e_{63}: overskirt), MEGCF computes a significantly higher score for e_{78} than for e_{14} and e_{63}, which is consistent with the fact that the images of all the three items contain e_{78}, *i.e.,* this user is more interested in the "backpack". This result illustrates that incorporating multimodal semantic entities can effectively model user preference over multimodal latent space.
- The similarity score between items i_{8261} and i_{3125} is higher than that between items i_{6339} and i_{3125}, *i.e.*, compared with i_{6339}, i_{8261} is more similar to i_{3125}, which also corresponds to the fact that i_{8261} and i_{3125} share three semantic entities, while i_{8261} and i_{6339} share only two. This result illustrates that capturing multimodal semantic correlation can better mine the similarity between items at the multimodal level.
- Based on this simple case, we find that there is still room for enhancing the multimodal semantic entity extraction in MEGCF. Specifically, since the deep method we leverage for visual semantic entity mining is pre-trained on the ImageNet dataset with only 1,000 categories, the extracted semantic entities are limited by these 1,000 categories. For example, all the styles of hats and bags in the images are roughly identified as "cowboy hat" and "backpack", respectively. In addition, directly transferring the deep approach from the

computer vision research field to the recommendation scenario would significantly reduce the accuracy of the semantic entity recognition. Nevertheless, MEGCF still achieves the state-of-the-art performance. Based on this analysis, MEGCF will be significantly enhanced if the categories of entities can be refined and the accuracy of multimodal semantic entity extraction can be improved.

5 CONCLUSION AND FUTURE WORK

In this work, we propose a novel GCN-based multimodal recommender method, referred to as MEGCF, which introduces multimodal semantic correlation to tackle the mismatch problem in multimodal recommendation scenarios. By constructing a symmetric linear graph convolution network, MEGCF can achieve simultaneous capture of high-order multimodal semantic correlation and collaborative signal. In addition, we design a review-based sentiment weighting strategy to enhance the neighbor aggregation in GCN-based methods, in order to better capture high-order structural features on the graph. We conduct extensive experiments on three real-world datasets. The obtained results demonstrate the state-of-the-art performance of MEGCF. Further ablation experiments and analysis validate the effectiveness and rationality of MEGCF.

The multimodal entity extraction and semantic correlation modeling in MEGCF still has room for improvement as follows:

- Improving the accuracy of the extraction of multimodal semantic entities can better model semantic correlation and thus reduce the negative impact of misidentified entities. Consequently, in future work, we aim at using a larger dataset to enhance the pre-training of the feature extraction module, while employing contrastive learning [45] techniques to improve the representation learning of multimodal features in a self-supervised manner.
- Multimodal semantic entity extraction in MEGCF fails to extract complete preference-related semantic information, thus leading to the limitation of low utilization of multimodal features. To tackle this problem, we aim at using techniques such as causal inference [37] and transformer [32] in order to discover and distinguish more fine-grained preference-related multimodal information in future research.
- Rich semantic information in multimodal content can also be used to enhance the interpretability of recommender systems, which is left for future work.

REFERENCES

[1] Rianne van den Berg, Thomas N. Kipf, and Max Welling. 2017. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263* (2017).

[2] Lei Chen, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. 2020. Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7–12, 2020*. AAAI Press, 27–34.

[3] Tianqi Chen, Weinan Zhang, Qiuxia Lu, Kailong Chen, Zhao Zheng, and Yong Yu. 2012. SVDFeature: A toolkit for feature-based collaborative filtering. *The Journal of Machine Learning Research* 13, 1 (2012), 3619–3622.

[4] Soheil Danesh, Tamara Sumner, and James H. Martin. 2015. SGRank: Combining statistical and graphical methods to improve the state of the art in unsupervised keyphrase extraction. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*. Association for Computational Linguistics, Denver, Colorado, 117–126.

[5] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 249–256.

[6] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A factorization-machine based neural network for CTR prediction. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19–25, 2017*. ijcai.org, 1725–1731.

[7] William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4–9, 2017, Long Beach, CA, USA*. 1024–1034.

[8] Ruining He and Julian J. McAuley. 2016. VBPR: Visual Bayesian personalized ranking from implicit feedback. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12–17, 2016, Phoenix, Arizona, USA*. AAAI Press, 144–150.

[9] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25–30, 2020*. ACM, 639–648.

[10] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. 2018. Adversarial personalized ranking for recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08–12, 2018*. ACM, 355–364.

[11] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3–7, 2017*. ACM, 173–182.

[12] Xiaotian Jiang, Zhendong Niu, Jiamin Guo, Ghulam Mustafa, Zi-Han Lin, Baomi Chen, and Qian Zhou. 2013. Novel boosting frameworks to improve the performance of collaborative filtering. In *Asian Conference on Machine Learning, ACML 2013, Canberra, ACT, Australia, November 13–15, 2013 (JMLR Workshop and Conference Proceedings)*, Vol. 29. JMLR.org, 87–99.

[13] Santosh Kabbur, Xia Ning, and George Karypis. 2013. Fism: Factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 659–667.

[14] Wang-Cheng Kang, Chen Fang, Zhaowen Wang, and Julian J. McAuley. 2017. Visually-aware fashion recommendation and design with generative image models. In *2017 IEEE International Conference on Data Mining, ICDM 2017, New Orleans, LA, USA, November 18–21, 2017*. IEEE Computer Society, 207–216.

[15] Dong Hyun Kim, Chanyoung Park, Jinoh Oh, Sungyoung Lee, and Hwanjo Yu. 2016. Convolutional matrix factorization for document context-aware recommendation. In *Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, September 15–19, 2016*. ACM, 233–240.

[16] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings*.

[17] Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings*. OpenReview.net.

[18] Yehuda Koren. 2008. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 426–434.

[19] Chenyi Lei, Dong Liu, Weiping Li, Zheng-Jun Zha, and Houqiang Li. 2016. Comparative deep learning of hybrid representations for image recommendations. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016*. IEEE Computer Society, 2545–2553.

[20] Chenxi Liu, Barret Zoph, Maxim Neumann, Jonathon Shlens, Wei Hua, Li-Jia Li, Li Fei-Fei, Alan L. Yuille, Jonathan Huang, and Kevin Murphy. 2018. Progressive neural architecture search. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part I (Lecture Notes in Computer Science)*, Vol. 11205. Springer, 19–35.

[21] Kang Liu, Feng Xue, Xiangnan He, Dan Guo, and Richang Hong. 2022. Joint multi-grained popularity-aware graph convolution collaborative filtering for recommendation. *IEEE Transactions on Computational Social Systems* (2022), 1–12. https://doi.org/10.1109/TCSS.2022.3151822

[22] Kang Liu, Feng Xue, and Richang Hong. 2022. RGCF: Refined graph convolution collaborative filtering with concise and expressive embedding. *Intelligent Data Analysis* 26, 2 (2022), 427–445.

[23] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 188–197.

[24] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI'09)*. AUAI Press, Arlington, Virginia, USA, 452–461.

[25] Ruslan Salakhutdinov and Andriy Mnih. 2007. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3–6, 2007*. Curran Associates, Inc., 1257–1264.

[26] Xiaoyuan Su and Taghi M. Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence* 2009 (2009).

[27] Rui Sun, Xuezhi Cao, Yan Zhao, Junchen Wan, Kun Zhou, Fuzheng Zhang, Zhongyuan Wang, and Kai Zheng. 2020. Multi-modal knowledge graphs for recommender systems. In *CIKM'20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19–23, 2020*. ACM, 1405–1414.

[28] Zhulin Tao, Yinwei Wei, Xiang Wang, Xiangnan He, Xianglin Huang, and Tat-Seng Chua. 2020. MGAT: Multimodal graph attention network for recommendation. *Inf. Process. Manag.* 57, 5 (2020), 102277.

[29] Hao Tian, Can Gao, Xinyan Xiao, Hao Liu, Bolei He, Hua Wu, Haifeng Wang, and Feng Wu. 2020. SKEP: Sentiment knowledge enhanced pre-training for sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5–10, 2020*. Association for Computational Linguistics, 4067–4076.

[30] Quoc-Tuan Truong and Hady W. Lauw. 2019. Multimodal review generation for recommender systems. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13–17, 2019*. ACM, 1864–1874.

[31] Quoc-Tuan Truong, Aghiles Salah, and Hady W. Lauw. 2021. Multi-modal recommender systems: Hands-on exploration. In *RecSys'21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*. ACM, 834–837.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4–9, 2017, Long Beach, CA, USA*. 5998–6008.

[33] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

[34] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018. RippleNet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22–26, i2018*. ACM, 417–426.

[35] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge graph convolutional networks for recommender systems. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13–17, 2019*. ACM, 3307–3313.

[36] Suhang Wang, Yilin Wang, Jiliang Tang, Kai Shu, Suhas Ranganath, and Huan Liu. 2017. What your images reveal: Exploiting visual contents for point-of-interest recommendation. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3–7, 2017*. ACM, 391–400.

[37] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2021. Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue. In *SIGIR'21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11–15, 2021*. ACM, 1288–1297.

[38] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4–8, 2019*. ACM, 950–958.

[39] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 165–174.

[40] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled graph collaborative filtering. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1001–1010.

[41] Yinwei Wei, Xiang Wang, Xiangnan He, Liqiang Nie, Yong Rui, and Tat-Seng Chua. 2021. Hierarchical user intent graph network for multimedia recommendation. *IEEE Transactions on Multimedia* (2021), 1–12. https://doi.org/10.1109/TMM.2021.3088307

[42] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, and Tat-Seng Chua. 2020. Graph-refined convolutional network for multimedia recommendation with implicit feedback. In *MM'20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12–16, 2020*. ACM, 3541–3549.

[43] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21–25, 2019*. ACM, 1437–1445.

[44] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International Conference on Machine Learning*. PMLR, 6861–6871.

[45] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *SIGIR'21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11–15, 2021*. ACM, 726–735.

[46] Le Wu, Xiangnan He, Xiang Wang, Kun Zhang, and Meng Wang. 2022. A Survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2022), 1–20. https://doi.org/10.1109/TKDE.2022.3145690

[47] 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 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21–25, 2019*. ACM, 235–244.

[48] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. 2018. Representation learning on graphs with jumping knowledge networks. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10–15, 2018 (Proceedings of Machine Learning Research)*, Vol. 80. PMLR, 5449–5458.

[49] Feng Xue, Xiangnan He, Xiang Wang, Jiandong Xu, Kai Liu, and Richang Hong. 2019. Deep item-based collaborative filtering for top-N recommendation. *ACM Transactions on Information Systems (TOIS)* 37, 3 (2019), 1–25.

[50] Y. Koren, R. Bell, and C. Volinsky. 2009. Matrix factorization techniques for recommender systems. *IEEE Computer* 42, 8 (2009), 30–37.

[51] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 974–983.

[52] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 353–362.

[53] Lei Zheng, Vahid Noroozi, and Philip S. Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6–10, 2017*. ACM, 425–434.

Received 10 December 2021; revised 14 April 2022; accepted 30 May 2022