Information Processing and Management 57 (2020) 102277

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

MGAT: Multimodal Graph Attention Network for Recommendation

Zhulin Tao[a], Yinwei Wei[b], Xiang Wang[c,*], Xiangnan He[d], Xianglin Huang[*,a], Tat-Seng Chua[c]

[a] State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing, China
[b] Shangdong University, China
[c] National University of Singapore, Singapore
[d] University of Science and Technology of China, Hefei, China

ARTICLE INFO

Keywords:
Personalized recommendation
Graph
Gate mechanism
Attention mechanism
Micro-videos

ABSTRACT

Graph neural networks (GNNs) have shown great potential for personalized recommendation. At the core is to reorganize interaction data as a user-item bipartite graph and exploit high-order connectivity among user and item nodes to enrich their representations. While achieving great success, most existing works consider interaction graph based only on ID information, foregoing item contents from multiple modalities (*e.g.,* visual, acoustic, and textual features of micro-video items). Distinguishing personal interests on different modalities at a granular level was not explored until recently proposed MMGCN (Wei et al., 2019). However, it simply employs GNNs on parallel interaction graphs and treats information propagated from all neighbors equally, failing to capture user preference adaptively. Hence, the obtained representations might preserve redundant, even noisy information, leading to non-robustness and suboptimal performance. In this work, we aim to investigate how to adopt GNNs on multimodal interaction graphs, to adaptively capture user preference on different modalities and offer in-depth analysis on why an item is suitable to a user. Towards this end, we propose a new Multimodal Graph Attention Network, short for MGAT, which disentangles personal interests at the granularity of modality. In particular, built upon multimodal interaction graphs, MGAT conducts information propagation within individual graphs, while leveraging the gated attention mechanism to identify varying importance scores of different modalities to user preference. As such, it is able to capture more complex interaction patterns hidden in user behaviors and provide a more accurate recommendation. Empirical results on two micro-video recommendation datasets, Tiktok and MovieLens, show that MGAT exhibits substantial improvements over the state-of-the-art baselines like NGCF (Wang, He, et al., 2019) and MMGCN (Wei et al., 2019). Further analysis on a case study illustrates how MGAT generates attentive information flow over multimodal interaction graphs.

1. Introduction

The personalized recommendation has become a key component in many user-oriented services, filtering items of interest for users accurately and timely, for series such as products in E-commerce (*e.g.,* Amazon and Taobao), friends in social networking (*e.g.,*

* Corresponding author.
E-mail addresses: taozhulin@gmail.com (Z. Tao), weiyinwei@hotmail.com (Y. Wei), xiangwang@u.nus.edu (X. Wang), xiangnanhe@gmail.com (X. He), huangxl@cuc.edu.cn (X. Huang), chuats@comp.nus.edu.sg (T.-S. Chua).

https://doi.org/10.1016/j.ipm.2020.102277
Received 7 January 2020; Received in revised form 15 April 2020; Accepted 20 April 2020

Facebook and WeChat), and micro-videos in content sharing (*e.g.,* Tiktok and Kwai) platforms. It is hence of crucial importance to predict how likely a user would interact with an item(*e.g.,* purchase, click, and view). Towards this end, existing recommender models (Chen et al., 2012; He & Chua, 2017; He et al., 2017; Koren, Bell, & Volinsky, 2009; Rendle, Freudenthaler, Gantner, & Schmidt-Thieme, 2009; Wang, He, Wang, Feng, & Chua, 2019) mainly focus on exploiting historical user-item interactions to perform the predictions. These models largely follow a general paradigm equipped with two key components — (1) representation learning, which transforms each user-item pair and its side information to appropriate representations, and (2) interaction modeling, which performs predictions based on the representations. For example, as an early work, matrix factorization (MF) (Koren et al., 2009; Rendle et al., 2009) merely projects ID of an user (or an item) to his/her embedding; later on, FISM (Kabbur, Ning, & Karypis, 2013) and SLIM (Ning & Karypis, 2011) average the embeddings of the historical items as the representation of a user; moreover, SVD++ (Chen et al., 2012) aggregates ID embeddings of a user and historical items together, while NAIS (He, He, et al., 2018) adopts attention mechanism over historical items to achieve better performance. Clearly, the representation quality serves as a key influential factor in the effectiveness of recommender models.

Recent studies (van den Berg, Kipf, & Welling, 2017; Wang, He, Cao, Liu, & Chua, 2019; Wang, He, Wang, et al., 2019; Zheng, Lu, Jiang, Zhang, & Yu, 2018) have shown that, adopting graph neural networks (GNNs) is able to augment representation learning with high-order relationships among users and items. These models organize interactions among users and micro-videos as bipartite graphs, which exhibit their relationships as their connectivity. In the bipartite graphs, the first-order connectivity (i.e., direct connections) presents the pre-existing features of users and items (e.g., historical items are seen as user profile), while higher-order connectivity reflects behavioral similarity among users, audience similarity among items, and collaborative filtering signals. Such high-order connectivity is useful to enrich the representations of users and items. Meanwhile, inspired by the information propagation of GNNs, these recommenders adopt the same idea to refine representations of users and items — that is, they first generate the message being passed from each neighbor, then aggregate messages from all neighbors to update the embeddings of user and item nodes, and recursively perform such propagation to consider high-hop neighbors. For example, GC-MC (van den Berg et al., 2017), NGCF (Wang, He, Wang, et al., 2019), and LightGCN (He et al., 2020) benefit from such propagation, wherein NGCF encodes collaborative filtering (CF) signals into representations and achieves state-of-the-art performance, and LightGCN further simplifies the neural network design of NGCF and shows better CF effectiveness. Moreover, GNN-based recommenders have shown great potential in many challenging scenarios, ranging from social (Fan et al., 2019; Wu, Sun, et al., 2019; Wu, Zhang, et al., 2019), session-based (Song et al., 2019; Wu et al., 2019c; Zheng, Gao, He, Li, & Jin, 2020) to knowledge graph-based (Wang, Zhao, Xie, Li, & Guo, 2019;Wang, He, Cao, et al., 2019) recommendation.

While achieving great success, these methods are not sufficient to establish satisfactory representations for items with multimodal contents, such as movies in Netflix and micro-videos in Tiktok, which typically involve visual, acoustic, and textual contents (Wei et al., 2019). One key reason is that the GNN-based representation learning lacks explicit modeling of modality difference, which is latent in user-item interactions and crucial to propagate personal interests at the granularity of modalities. More specifically, most existing methods either build one interaction graph by viewing multimodal contents as node features (van den Berg et al., 2017), or analyse multiple interaction graphs based in parallel on individual modalities (Wei et al., 2019), without disentangling user tastes on different modalities,which may have different influence to information propagation over graphs. Taking micro-video recommendation as an example, a user may prefer micro-videos with the same BGM (background music) fitting the mood of the scenes, while the scenes might be visually different; or, she may prefer the BGM of some micro-videos, while caring more about the visual scene of other micro-videos. As a result, homogenizing or unifying multimodal channels is insufficient to identify varying importance of modalities, hindering the information propagation and resulting in suboptimal representations.

In this work, we aim to investigate how to leverage GNNs properly and effectively propagate information over multimodal interaction graphs, while capturing user preference on different modalities. Towards this end, we propose a new Multimodal Graph Attention Network, termed as MGAT, which is equipped with three designs: (1) multimodal interaction graph construction for capturing users fine-grained preference w.r.t. different modalities, which follows the prior effort (MMGCN Wei et al. (2019)) to establish parallel graphs among user and item nodes; (2) embedding propagation on single graph for encoding behavioral patterns of users into representations of users and items, which updates the representation of a user (or item) based on the user's historical items (or its user group); and (3) gated attention aggregation across graphs for identifying varying importance of different modalities, which exploits the other modalities to learn the weights of each neighbor and then guide a propagation. As a result, such attention mechanism endows MGAT with the ability to disentangle personal interests at the granularity of modality. Moreover, when recursively performing such information propagation, we can obtain an information flow from higher-order neighbors and reasonably explore user interests based on the attentions. We conduct extensive experiments on two real-world datasets, Tiktok and MovieLens, to demonstrate the rationality and effectiveness of our MGAT model.

Note that a preliminary version of this work has been published as a conference paper in ACM MM 2019 (Wei et al., 2019). We summarize the key enhancements as follows:

- We have enhanced the framework of MMGCN as MGAT. Refer to MMGCN, which utilizes original GCN that may cause overlap information conflict and adopts the highest order representation for prediction, MGAT adopts GNN to aggregate the information from neighbor nodes and combine the aggregated result with the information of the head entity node. Meanwhile, MGAT concatenates different orders' representations of nodes to distinguish variant contributions from different orders to the interaction prediction.
- In MGAT, We introduce the gated attention mechanism to control and weight the information flow propagation in each layer of each modality.

- We complement all the experiments except baselines to justify the effectiveness of our reconstructed model and proposed mechanism.
- We reorganized the paper to emphasize the motivation of this extended version.

In a nutshell, the key contributions are summarized as follows:

- We develop a new method MGAT, which incorporates attention mechanism into the graph neural network framework, to disentangle user preferences on different modalities.
- Technically, the model introduces the gated attention mechanism to control and weight the information flow in multimodal interaction graphs, which facilitates the understanding of user behaviors.
- We perform extensive experiments on two datasets to verify the rationality and effectiveness of MGAT. Moreover, because of user privacy, only user IDs are considered in this work. We will release the code and parameter settings upon acceptance.

2. Related works

2.1. Multi-modal personalized recommendation

Personalized recommendation systems have been successfully applied to many applications, such as e-commerce, news, and social media platforms. Typically, Most existing approaches adopt the CF-based method (He, Du, et al., 2018; 2017; Rendle, 2010; Wang, He, Feng, Nie, & Chua, 2018; Wang, He, Nie, & Chua, 2017; Zhang et al., 2016)(Tao, Wang, He, Huang, & Chua, 2019). Recently, with the success of deep neural network(DNN) in computer vision,acoustic, and natural language processing task (Hong et al., 2017; Hong, Yang, Wang, & Hua, 2015; Liu, Nie, Wang, Tian, & Chen, 2019; Nie et al., 2017; Wong, Chen, Mau, Sanderson, & Lovell, 2011), DNN is also introduced into the multimodal domain (Nie et al., 2017; Wang et al., 2012). In particular, some efforts have been dedicated to integrate the item features, which are extracted from multiple modalities by the pre-trained deep learning models, into the CF based model for enhancing the item representations. For instance, *Chen* et al. Chen, He, and Kan (2016) proposed a model named CITING, which mines and fuses textual features to model the semantic of social media images for image tweet recommendation. Covington, Adams, and Sargin (2016) developed a two-stage model that is composed of the deep generation model and ranking model for video recommendation. Gao, Zhang, and Xu (2017) developed a dynamic recurrent neural network which fuses video semantics and user interests to model users' dynamic preferences. Different from these approaches, we focus on modeling and disentangling user preferences on different modalities.

2.2. Graph convolution network

Due to its effectiveness and simplicity, graph convolution networks are widely used in various applications (van den Berg et al., 2017; Hamilton, Ying, & Leskovec, 2017; Niepert, Ahmed, & Kutzkov, 2016; Perozzi, Al-Rfou, & Skiena, 2014). With graph convolutional operations, the nodes' local structure information can be encoded into their representations by massage passing and aggregation. Specifically, PinSAGE (Hamilton et al., 2017) is the first GCN-based model that has been successfully applied in industry, which generates the node's representation by sampling and aggregating its neighbors' features. Meanwhile, a lot of attention is also paid to the graph-based recommendation methods (Cao, Wang, He, Hu, & Chua, 2019; Wang et al., 2019d)(Wang, Jin, et al., 2020; Wang, Xu , et al., 2020). NGCF (Wang, He, Wang, et al., 2019) explicitly integrates the collaborative signals into the embedding process and leads to the expressive modeling of high-order feature interaction in the bipartite graph. LightGCN (He et al., 2020) further simplifies the design of NGCF, so that it becomes easier to train and achieves better performance. MMGCN (Wei et al., 2019) tries to model the user preferences for different modalities on the model-specific user-item bipartite graph.

2.3. Gated attention mechanisms

To control the information propagation operations, the gate mechanism is introduced into some machine learning schemes. In LSTM (Hochreiter & Schmidhuber, 1997), the mechanism is used to implement the input-gate, forget-gate, and output-gate, which are used to memorize, forget and expose the memory content, respectively. Further, with the help of gate mechanism, GRU (Cho et al., 2014) adaptively resets or updates its memory content in the recurrent networks. Inspired by the gated mechanism in these methods, several graph-based models utilize the mechanism to improve their performance. For example, Li, Tarlow, Brockschmidt, and Zemel (2016) devised a novel graph-based model, dubbed GGCN, to learn the long-distance relations between nodes. Similar with the gated mechanism, the attention mechanism is also adopted by some modules to weight the importance of propagated information. AFM (Xiao et al., 2017) adopts an attention network to weight the importance of second-order cross features. Chen et al. (2017) proposed an attentive collaborative filtering framework, where the attention is used to mark the feature's significance at two levels for representing users' preferences. Considering the different neighbors' importance in graph convolutional operations, Velickovic et al. (2017) devised a graph attention network, which leveraged the multi-head attention to control the message passing. In our model, we propose a gated attention mechanism that utilizes the advantage of the gate and attention mechanisms to control and weight the information propagation.

3. Task formulation

Here we first introduce some key concepts used in our model, before presenting the task formulation.

- **User-Item Bipartite Graph:**Such interaction graph is built upon the user and item nodes, where the edges is constructed by the historical interactions between users and items. We formulate the graph as $G = (V, E)$, where $V = U \cup I$ is the node set involving U and I separately as the sets of user and item, and $E = \{(u, i)|u \in U, i \in I\}$ is the edge set, each of which represents the interaction between user u and item i. Note that we formalize G as a undirected graph.
- **Multimodal Interaction Graph:**Besides the interaction data, a multimedia item (*e.g.,* micro-video) is typically associated with ID and multimodal contents (*e.g.,* visual, acoustic, and textual features), while a user is assigned with ID information merely for simplicity. For each modality, an bipartite graph is devised between user and item nodes, where edges indicate interaction data. More formally, we present the multimodal interaction graph as a set $\{G_m\}$, where $m \in \{1, 2, 3\}$ denoting the visual, acoustic, and textual modalities, respectively.Moreover, we only adopt user IDs for the reason of user privacy
- **High-order Connectivity:**Inspired by recent GNN-based recommendation methods (Wang, He, Wang, et al., 2019; Wei et al., 2019), user-item relationships hidden in user behavior data can be explicitly formulated as their connectivity. In particular, the first-order connectivity of a user is composed of the user's historical items, directly profiling his/her preference; analogously, the user group of an item can act as the descriptive features. Furthermore, by conducting random walk on the graph, we can obtain the higher-order connections among nodes (*i.e.,* paths), which encode the collaborative signals. Taking the path $i_1 \rightarrow u_1 \rightarrow i_2 \rightarrow u_3$ as an example, we might conclude that u_1 and u_3 are likely to have similar preference, due to their similar behaviors in adopting i_2; we could further recommend i_1, that is selected by u_1 to u_3, which reflects the collaborative filtering effect.
- **Task Description:** We now formulate the task as follows:
  - **Input:** Multimodal interaction graphs, each node of which is associated with visual, acoustic and textual features.
  - **Output:** A recommender function that predicts the likelihood of interaction between an user u and an item i.

4. Methods

We here present a new model named Multimodal Graph Attention Network (MGAT). Fig. 1 demonstrates the overall framework of MGAT, which consists of four components: (1) embedding layer, which initializes ID embeddings of users and items; (2) embedding propagation layer on single-modal interaction graph, which performs the message-passing mechanism to capture user preferences on individual modalities; (3) gated attention aggregation across multimodal interaction graphs, which exploits the correlations with the other modalities to learn the weights of each neighbor in order to guide a propagation; and (4) prediction layer, which estimates the likelihood of an interaction based on the final representations.

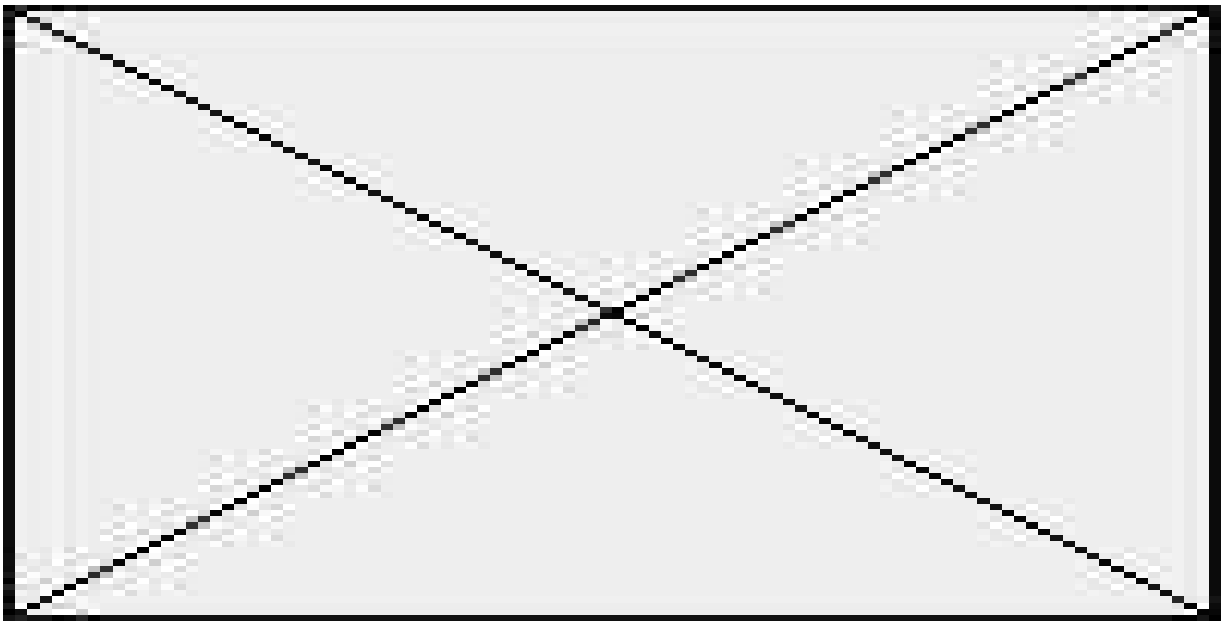

Fig. 1. On the left is the framework of our model, in which the gated attention-mechanism is incorporated in the information propagation process, and the **R** in the prediction layer is a Sigmoid function; on the right is an illustration of the gated attention-GNN structure, wherein f_a is attention mechanism and f_g is gate mechanism.

4.1. Embedding layer

Typically, a user and an item are associated with IDs. A widely-used solution to represent such ID information is to embed ID as a vectorized representation. In particular, user u and item i are separately projected as $\mathbf{e}_u$ and $\mathbf{e}_i$, which memorizes their general characteristics. Moreover, within individual interaction graphs, each item i has a pre-existing feature, $\mathbf{e}_{mi}$, to highlight its characteristics in the m-th modality. In addition, we assign an additional embedding $\mathbf{e}_{mu}$ with user u to capture the user preference in the m-th modality. All the embeddings are summarized as follows:

$$\mathbf{E} = \{\mathbf{e}_u, \mathbf{e}_i, \mathbf{e}_{m,u}, \mathbf{e}_{m,i} | u \in U, i \in I, m \in M\}, \tag{1}$$

where $\mathbf{e}_u, \mathbf{e}_{m,u} \in \mathbb{R}^{|U| \times d}$ and $\mathbf{e}_i, \mathbf{e}_{m,i} \in \mathbb{R}^{|I| \times d}$; N and M denote the numbers of users and items, respectively; and d is the embedding size. It is worth noting that, $\mathbf{e}_i$, $\mathbf{e}_u$, and $\mathbf{e}_{m,u}$ are randomly initialized and trained during optimization, while $\mathbf{e}_{m,i}$ is derived from fixed features via a trainable neural network.

4.2. Information propagation layer

Distinct from traditional recommender models that directly feed a pair of user and item embeddings into a prediction model, GNN-based methods exploit the interaction graph to refine the representations. However, studies on multimodal interaction graphs are under explored, and existing methods failed to disentangle user preference at the granularity of modality. Towards this end, we introduce the gated attention mechanism into information propagation.

4.2.1. Information aggregation

For an individual interaction graph, considering a ego node h with the first-hop neighbors $N_h = \{t | (h, t) \in E\}$, we formalize the information being propagated from such neighbors to h as follows:

$$\mathbf{e}_{m,N_h} = \text{LeakyReLU}\Big(\sum_{t \in N_h} f_a(h, t) f_g(h, t) W_{m,1} \mathbf{e}_{m,t} \Big), \tag{2}$$

where m is the modality indicator; $f_g(h, t)$ and $f_a(h, t)$ separate the roles of gate and attention components in the gated attention network; and $\mathbf{W}_{m,1}$ is a trainable weight matrix to distill useful clues. To be more specific, $f_g(h, t)$ is the propagation gate for deciding whether the information would be propagated from t to h. Meanwhile, $f_a(h, t)$ is the attention score indicating the contributions of t. In what follows, we elaborate these two components.

4.2.2. Propagation gate

Inspired by the gate mechanism originally adopted in GRU (Cho et al., 2014), the previous work GGCN (Li et al., 2016) employs similar components to decide whether a neighbor would propagate information to the ego. Inspired by GGCN, we utilize such gate mechanism to control the information flow when performing a propagation. We implement $f_g(h, t)$ using the following three types of gates:

- *Inner Product Gate* that calculates the inner-product of $\mathbf{e}_{m,h}$ and $\mathbf{e}_{m,t}$ first, and uses $\frac{1}{\sqrt{d}}$ to handle the varying number of neighbors, which is formalized as:

$$f_{g_i}(h, t) = \delta\left(\frac{\mathbf{e}_{m,h}^{\top} \mathbf{e}_{m,t}}{\sqrt{d}}\right), \tag{3}$$

  where $\delta(.)$ is the sigmoid function, and d is the out degree of t. Such gate is dependent on the affinity between h and t.
- *Concatenation Gate* that concatenates two representations, followed by a linear transformation:

$$f_{g_c}(h, t) = \delta\left(\frac{W_c(\mathbf{e}_{m,h} || \mathbf{e}_{m,t})}{\sqrt{d}}\right), \tag{4}$$

  where $||$ represents the concatenation operator, W_c is trainable weight matrix.
- *Bi-interaction Gate* that combines these two kinds of gates and endows such mechanism with more flexibility, which is formulated as:

$$f_{g_b}(h, t) = \delta\left(\frac{W_b(\mathbf{e}_{m,h} || \mathbf{e}_{m,t}) + \mathbf{e}_{m,h} \odot \mathbf{e}_{m,t}}{\sqrt{d}}\right), \tag{5}$$

  where $\odot$ is the element-wise multiplication operation.

As such, the gate scores within a single-modal interaction graph reflect whether the specific modality plays a role when profiling user preference.

4.2.3. Neighbor-ware attention

Thereafter, we also introduce the attention mechanism to learn the varying importance of each neighbor, as follows:

$$f_a(h, t) = (W_{m,h}\mathbf{e}_{m,h})^{\top}\tanh(W_{m,t}\mathbf{e}_{m,t}), \tag{6}$$

where tanh is used as a nonlinear activation function; and $W_{m,h}$ and $W_{m,t}$ are the learnable transformation matrices. For simplicity we consider the inner product here to obtain the attention weights, which reflects the affinity between the two nodes. Hereafter, we employ the softmax function to normalize the attention weights across all neighbors, which is formulated as follows:

$$f_a(h, t) = \frac{\exp f_a(h, t)}{\sum_{t' \in N_h} \exp f_a(h, t')}, \tag{7}$$

where the final attention scores are able to distinguish varying importance scores of neighbors.

Having obtained the gate and attention scores, we conduct their product $f_g(h, t)f_a(h, t)$, so as to propagate personal interests at the granularity of modalities. To be more specific, $f_g(h, t)$ determines whether the items within individual modalities would propagate the information to the target users, while $f_a(h, t)$ discovers the varying contributions of these items on user representations.

4.2.4. Information combination

We then utilize the information being propagated from neighbors $\mathbf{e}_{m,N_h}$ to update the representation of node h. In particular, the ID embeddings of node h, $\mathbf{e}_h$, is treated as the anchor across modalities, serving as the highway to perform cross-modality propagation. Hence, we first formulate the process as:

$$\tilde{\mathbf{e}}_{m,h} = \text{LeakyReLU}(\mathbf{W}_{m,2}\mathbf{e}_{m,h} + \mathbf{e}_h), \tag{8}$$

where $\mathbf{W}_{m,2}$ is the transformation matrix;and $\mathbf{e}_h$ in essence functions as the virtual supernode connecting $\{\mathbf{e}_{m,h}\}$, $\forall m \in M$. Thereafter, we combine $\tilde{\mathbf{e}}_{m,h}$ with $\mathbf{e}_{m,N_h}$ as follows:

$$\mathbf{e}^{(1)}_{m,h} = \text{LeakyReLU}\left(\mathbf{W}_{m,3}\mathbf{e}_{m,N_h} + \tilde{\mathbf{e}}_{m,h}\right), \tag{9}$$

which $\mathbf{e}^{(1)}_{m,h}$ denotes the representation of node h after encoding the first-order connectivity; and $\mathbf{W}_{m,2}$ is the trainable weight matrix.

4.2.5. High-order propagation

Following the prior efforts (Wang, He, Cao, et al., 2019; Wang, He, Wang, et al., 2019; Wei et al., 2019), we can stack more information propagation layers to exploit the higher-order connectivity among nodes and further enrich the representations. More formally, the representation of node h is recursively defined as:

$$\mathbf{e}^{(l)}_{m,h} = \text{LeakyReLU}\left(\mathbf{W}^{(l-1)}_{m,3}\mathbf{e}^{(l-1)}_{m,N_h} + \tilde{\mathbf{e}}^{(l-1)}_{m,h}\right), \tag{10}$$

where $\mathbf{e}^{(l-1)}_{m,h}$ is the representation after $(l-1)$ propagation steps, storing the information from the $(l-1)$-hop neighbors; and $\mathbf{e}^{(0)}_{m,h}$ is the initial embedding $\mathbf{e}_{m,h}$.

After updating the representations of nodes in the specific modality m, we can combine the representations from different modalities into a new representation which can be formulated as:

$$e^{(l)}_h = \frac{1}{|M|}\sum_{m \in M} e^{(l)}_{m,h}. \tag{11}$$

4.3. Prediction layer

Assuming that the number of information propagation is L, we can produce the final representations of nodes, emphasizing different orders of neighbors, as follows:

$$\mathbf{e}^*_u = \mathbf{e}^{(0)}_u \| \cdots \| \mathbf{e}^{(L)}_u, \qquad \mathbf{e}^*_i = \mathbf{e}^{(0)}_i \| \cdots \| \mathbf{e}^{(L)}_i, \tag{12}$$

Finally, we conduct inner product of user and item representations, so as to predict their matching score:

$$\hat{y}_{ui} = \mathbf{e}^{*\top}_u \mathbf{e}^*_i. \tag{13}$$

4.4. Optimization

Following the mainstream optimization method (He et al., 2017; Rendle et al., 2009; Wang, He, Wang, et al., 2019), we adopt Bayesian Personalized Ranking (BPR) to optimize the model parameters, which assumes that the user prefers items that were interacted before more than those without prior interactions. We can formulate this as follows:

$$\mathcal{L} = \sum_{(u,i,j) \in O} - ln(\delta(\hat{y}_{ui} - \hat{y}_{uj})) + \lambda \|\theta\|^2_2 \tag{14}$$

where $O \in \{(u, i, j)|(u, i) \in R^+, (u, j) \in R^-\}$ is the training dataset; R^+ is the dataset containing observed interaction between user u and item i, while R^- is the unobserved interactions; $\delta(.)$ is the sigmoid function; λ is the decay factor and θ is the parameters used in the model.

5. Experiments

In this section, we present our experimented in results detail, which include the experimental settings, performance comparison, and the case studies of MGAT.

5.1. Experiment settings

5.1.1. Datasets

To evaluate the performance of MGAT, we perform experiments on two publicly-accessible datasets, they are the MovieLens[1], Tiktok[2]. the two statistics of the two datasets are in Table 1., Their details are given below:

- **MovieLens Dataset:** This dataset has been widely used for the personalized recommendation, and it contains a series of subsets. In this work, we chose MovieLens-10M as the experimental dataset. Regarding the multi-modal feature extraction, we adopted a pre-trained ResNet50 (He, Zhang, Ren, & Sun, 2016) to extract visual features from the keyframes of videos, and the acoustic features were learned from audio trackers by VGGish (Hershey et al., 2017). Moreover, textual features were produced by Sentence2Vector (Arora, Liang, & Ma, 2016) from text content, which includes titles and descriptions.
- **Tiktok Dataset:** This dataset is published in a data mining competition by Tiktok, a popular micro-video sharing platform. It contains micro-videos with a duration of 3–15 seconds along with the textual video captions provided by the users. We used the original desensitized multi-modal feature vectors provided in this dataset. All the modality features are desensitized and provided in a vector manner without raw data. These videos are with a duration of 3–15 seconds, and the textual contents are derived from the captions given by users.

We randomly divided each of the datasets into three parts — training (80%), validation (10%), and test (10%) sets. The validation set is used to tune the hyper-parameters and we select the best-trained model. We report the final performance of the best-performing model on the test set.

5.1.2. Evaluation metrics

This section introduces the evaluation metrics and parameter settings used in our experiments. We adopt three widely used evaluation metrics: Precision@K, Recall@K, and NDCG@K. We set $K = 10$ and report the averaged performance achieved for all users in the test set. The negative items of each user are defined as those having no interactions with the user. The codes of all experiments are implemented using the PyTorch framework. For all models, the embedding size is 64 in all models, and the batch size is 1024. We adopt the Xavier initializer to initialize all the model parameters. Hereafter, we optimize all the models with Adam optimizer. Moreover, we apply grid search for hyper-parameters fining, where the values of the learning rate is selected from $\{1e-1, 1e-2, 1e-3, 1e-4, 1e-5\}$, and those for weight decay and the attention dropout ratio are chosen from $\{1e-1, 1e-2, 1e-3, 1e-4, 1e-5\}$, $\{0.1, 0.2, ..., 0.8\}$, respectively. Without specification, the node dropout and message dropouts are 0.0. The other baselines use the hyper-parameters as used in the original papers.

5.1.3. Baselines

To evaluate the performance of our model, we compare MGAT with the following the baselines:

- **VBPR** (He & McAuley, 2016). This model injects visual features into the representation of the item. Subsequently, it utilizes Matrix Factorization to learn the representations of users and items based on their historical interactions. In our experiment, we concatenate the multimodal features of the micro-video into a single feature vector. We then integrate it with ID information to predict the interaction between users and items.
- **ACF** (Chen et al., 2017). It introduces the item-level and component-level attention to handle the implicit feedback for the multimedia recommendation. In our experiment, we adopt a similar component-level attention mechanism for each modality for interaction prediction.
- **GraphSAGE** (Hamilton et al., 2017). It is a graph-based model that aggregates information from the neighbor nodes to represent the unseen data. In this work, we concatenate features of the three modalities to represent each of the nodes.
- **NGCF** (Wang, He, Wang, et al., 2019). NGCF integrates the collaborative signal into the embedding process in an explicit manner. It models the high-order feature interactions in the bipartite graph by incorporating information passing from multiple levels of neighbors. In this paper, we concatenate all of the multimodal features as the item representation, which are used in the subsequent process of embedding interactions in NGCF.
- **MMGCN** (Wei et al., 2019). MMGCN is a graph-based algorithm. To learn representations of user preferences on different modalities, it devises a model-specific bipartite graph based on the user-item interactions for each modality. After that, it aggregates all model-specific representations to obtain the representations of users or items for prediction.

[1] https://grouplens.org/datasets/movielens/.
[2] http://ai-lab-challenge.bytedance.com/tce/vc/.

Table 1
Statistics of the Tiktok and MovieLens datasets. Note that the notations of V, A, and T represent the number of features used for the raw visual, acoustic, and textual data, respectively.

Dataset	#Interactions	#Items	#Users	Sparsity	V	A	T
Tiktok	726,065	76,085	36,656	99.99%	128	128	128
MovieLens	1,239,508	5986	55,485	99.63%	2048	128	100

5.2. Performance comparison

All results of the experiments are as shown in Table 2, and we have the following observations:

- MGAT outperforms all the baseline models. It demonstrates the reasonable design of our model. Compared to traditional collaborative filtering methods (VBPR and ACF) that only consider the direct user-item connections, our MGAT uses high-order connectivity to facilitate the representation learning. Compared to GNN-based recommenders (GraphSage and NGCF) which uses one bipartite graph and simply unifies features from different modalities as one, MGAT identifies three channels to propagate useful signals, having better representation ability. Compared to MMGCN that applies post fusion to integrate representations of individual modalities, our MGAT discriminates the importance of each modality via the well-designed attention mechanism, so as to identify fine-grained preference of users.
- Both multimodal graph-based models outperform the other baselines. Comparing with the algorithms learning preferences from equal-weighted multimodal features, MGAT and MMGCN consistently achieve better performance. This demonstrates that incorporating attention to different modalities can help to model user preferences better.
- The performances of the graph-based models are better than the CF-based model on the Tiktok dataset. It verifies that injecting the information of neighbors into the node representation by message-passing can improve the representation of micro-videos. Moreover, the performances of GraphSAGE and NGCF are poorer than VBPR on the MoviLens. We attribute such findings to the data differences: as the videos in Tiktok dataset is shorter than that in MovieLens, the raw features of long videos would contain more complex information even noises, and the relationships among multiple modalities are highly entangled.
- Unexpectedly, the performance of ACF is poor in all experiments. It may be caused by the modification of the implementation of the ACF algorithm, in which we replaced the features by modal-specific features in component-level for a fair comparison.

5.3. Case studies for MGAT

In this section, we will present the case studies for MGAT to investigate the factors that may have effects on the performance of our model.

5.3.1. Effect of high-order connectivity

In this section, we evaluate how high-order connectivity affects the performance of MGAT. Specifically, We conducted experiments for three variants of MGAT, which incorporates different orders of neighbors for node representation. For example, MGAT_3 means using three orders of neighbors. Moreover, the embedding size is 64 on both datasets.

As shown in Table 3. The performance comparison among MGAT variants modeling different feature orders. Based on our experiments, MGAT_2 performs better than MGAT_1 and MGAT_3, which is consistent to the observations in NGCF (Wang, He, Wang, et al., 2019), MMGCN (Wei et al., 2019), GAT (Velickovic et al., 2017) papers. This indicates the over-smoothing issue of graph neural networks, that is, stacking more graph convolution layers or distilling signals from higher-order neighbors, easily introduces noises from remote neighbors and leads to suboptimal performance.

5.3.2. Effect of gate and attention

In our model, we introduce the gated attention mechanism to control and weight information propagation. To study its effects on the model, we conduct experiments on both datasets to evaluate the performance of the two-order MGAT model with different

Table 2
Performance Comparison between MGAT and the state-of-the-art recommendation algorithms on the Tiktok and MoviewLens datasets.

	Tiktok			MovieLens		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
VBPR	0.0972	0.4878	0.3136	0.1172	0.4724	0.2852
ACF	0.0873	0.4429	0.2867	0.1078	0.4304	0.2589
GraphSAGE	0.1028	0.4972	0.3210	0.1132	0.4532	0.2647
NGCF	0.1065	0.5008	0.3226	0.1156	0.4626	0.2732
MMGCN	**0.1164**	**0.5520**	**0.3423**	**0.1211**	**0.5138**	**0.3062**
MGAT	**0.1251**	**0.5965**	**0.3838**	**0.1272**	**0.5412**	**0.3251**
%Improv.	7.47%	8.06%	12.12%	5.03%	5.33%	6.17%

Table 3
Effect of different order.

	Tiktok			MovieLens		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
MGAT_1	0.121	0.5773	0.3681	0.1213	0.512	0.3054
MGAT_2	**0.1251**	**0.5965**	**0.3838**	**0.1272**	**0.5412**	**0.3251**
MGAT_3	0.1203	0.5775	0.3657	0.1242	0.5262	0.3145

combinations of the gating and attention mechanisms. MGAT_no means the base model without attention and gate mechanisms, MGAT_g denotes the base model without gate mechanism, and MGAT indicates the base model with the gated attention mechanism, The result is presented in Table 4.

As shown in Table 4, the performance of MGAT_g is better than MGAT_no, which verifies the performance gains by introducing the gate mechanism to control the propagated information from the neighbors. Meanwhile, the performance of MGAT is better than MGAT_g, which demonstrates that the gated attention mechanism can improve the performance of the model by weighting the propagated information. Moreover, MGAT_g is the variant discarding the attention mechanism only, while keeping the gate mechanism remained. Compared to MGAT_g, MGAT achieves better performance, indicating that the attention mechanism has positive effects.

5.3.3. Effect of different gate mechanisms

To evaluate different effects caused by different gated attention mechanisms, we conduct experiments on three variants of the gated attention mechanism, including the inner propagation gate (MGAT_i), the concatenation propagation gate (MGAT_c), and the Bi-interaction propagation gate (MGAT_bi).

As shown in the Table 5, the performance of MGAT_i outperforms the other two models. This indicates that the inner-product may more suitable to model information relations in our multimodal graph-based model. In contrast, the other models with the transform matrix may suffer from overfitting caused by the over-parameterized transform matrix.

5.4. A case study

As introduced in Section 4, our model adopts the gated attention mechanism, which is utilized to control and weight the information flow propagation. In Fig. 2, we sample ten neighbors of some node and then visualize the gated attention mechanism values for nodes. The rows denote the indexes of the neighbors, and the columns indicate the attention values of parameters. Moreover, the colors represent the value of different modalities.

As shown in Fig. 2, we observed that different nodes have different attention values produced by the gated attention mechanism in terms of modalities. This indicates that the importance of the neighbors is different. Besides, the attention values are also different across different modalities in most of the nodes; this demonstrates that the importance of features of different modalities is also different for the same node. Moreover, this mechanism makes MGAT to have better explainability, which is a common concern in recommender systems (Ren, Liang, Li, Wang, & de Rijke, 2017).

6. Conclusions

This paper presents a graph-based algorithm, named MGAT, which models user preferences with high-order neighboring information and the attention mechanism across different modalities. Specifically, in the process of modeling users' preferences, multimodal features are injected into the embeddings by the high order connectivities and message passing mechanism. Meanwhile, the gated attention mechanism is introduced to control and weight the propagated information from the high-hop neighbors. Extensive experiments conducted on two real-world datasets demonstrate the effectiveness of our model. In our model, we model users' preferences on different modalities, while there are more types of features in micro-videos, such as relations (Shang et al., 2019) and causality. Hence, we will pay more attention to understand micro-videos and will utilize more features for micro-video recommendation in future work.

Table 4
Effects of gate and attention mechanisms.

	Tiktok			MovieLens		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
MGAT_no	0.1198	0.5644	0.3621	0.1251	0.5323	0.3187
MGAT_g	0.1215	0.5748	0.3629	0.126	0.5344	0.319
MGAT	**0.1251**	**0.5965**	**0.3838**	**0.1272**	**0.5412**	**0.3251**

Table 5
Effect of different gate mechanisms.

	Tiktok			MovieLens		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
MGAT_c	0.1224	0.5866	0.3753	0.124	0.5263	0.3153
MGAT_i	**0.1251**	**0.5965**	**0.3838**	**0.1272**	**0.5412**	**0.3251**
MGAT_bi	0.1222	0.5887	0.3773	0.1249	0.527	0.3162

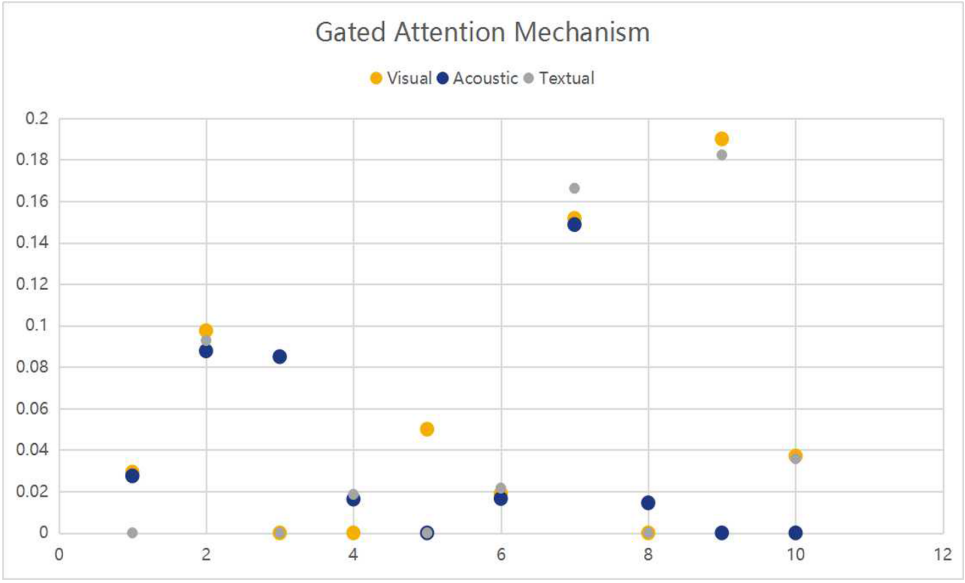

Fig. 2. Visualization of gated attention mechanism from Tiktok dataset.

CRediT authorship contribution statement

Zhulin Tao: Writing - original draft, Visualization, Investigation, Validation, Investigation, Software, Methodology, Conceptualization. **Yinwei Wei:** Resources, Data curation, Writing - review & editing, Supervision. **Xiang Wang:** Conceptualization, Writing - review & editing, Supervision. **Xiangnan He:** Writing - review & editing. **Xianglin Huang:** Writing - review & editing. **Tat-Seng Chua:** Writing - review & editing.

Acknowledgments

This research is part of NExT++ research and also supported by the National Research Foundation Singapore under its AI Singapore Programme, Linksure Network Holding Pte Ltd, and the Asia Big Data Association(Award No.: AISG-100E-2018-002). NExT++ is supported by the National Research Foundation, Prime Minister's Office, Singapore under its IRC@SG Funding Initiative. Moreover, this research is also supported by the national key research and development program of china(no.2019YFB1406201) and the future school program (no.CSDP17FS3231).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ipm.2020.102277 .

References

Arora, S., Liang, Y., & Ma, T. (2016). *A simple but tough-to-beat baseline for sentence embeddings. ICLR*1–16.
Tao, Z., Wang, X., He, X., Huang, X., & Chua, T. (2019). HoAFM: A High-order Attentive Factorization Machine for CTR Prediction. *Information Processing&Management,* 102076.
van den Berg, R., Kipf, T. N., & Welling, M. (2017). Graph convolutional matrix completion. *CoRR, abs/1706.02263*.
Cao, Y., Wang, X., He, X., Hu, Z., & Chua, T. (2019). *Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. WWW*151–161.
Chen, J., Zhang, H., He, X., Nie, L., Liu, W., & Chua, T. (2017). *Attentive collaborative filtering: Multimedia recommendation with item- and component-level attention. SIGIR*335–344.
Chen, T., He, X., & Kan, M. (2016). *Context-aware image tweet modelling and recommendation. ACM MM*1018–1027.

Chen, T., Zhang, W., Lu, Q., Chen, K., Zheng, Z., & Yu, Y. (2012). Svdfeature: A toolkit for feature-based collaborative filtering. *JMLR,* 3619–3622.
Cho, K., van Merrienboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). *Learning phrase representations using RNN encoder-decoder for statistical machine translation. EMNLP*1724–1734.
Covington, P., Adams, J., & Sargin, E. (2016). *Deep neural networks for youtube recommendations. RecSys*191–198.
Fan, W., Ma, Y., Li, Q., He, Y., Zhao, Y. E., Tang, J., & Yin, D. (2019). *Graph neural networks for social recommendation. WWW*417–426.
Gao, J., Zhang, T., & Xu, C. (2017). *A unified personalized video recommendation via dynamic recurrent neural networks. ACM MM*127–135.
Hamilton, W., Ying, Z., & Leskovec, J. (2017). *Inductive representation learning on large graphs. NIPS*1024–1034.
He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition. CVPR*770–778.
He, R., & McAuley, J. (2016). *Vbpr: Visual Bayesian personalized ranking from implicit feedback. AAAI*1–8.
He, X., & Chua, T. (2017). *Neural factorization machines for sparse predictive analytics. SIGIR*355–364.
He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020). Lightgcn: Simplifying and powering graph convolution network for recommendation. *CoRR, abs/2002.02126*.
He, X., Du, X., Wang, X., Tian, F., Tang, J., & Chua, T. (2018). *Outer product-based neural collaborative filtering. IJCAI*2227–2233.
He, X., He, Z., Song, J., Liu, Z., Jiang, Y., & Chua, T. (2018). NAIS: Neural attentive item similarity model for recommendation. *TKDE, 30*(12), 2354–2366.
He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. (2017). *Neural collaborative filtering. WWW*173–182.
Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, R. C., ... Seybold, B., et al. (2017). *CNN architectures for large-scale audio classification. ICASSP*. IEEE131–135.
Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.
Hong, R., Li, L., Cai, J., Tao, D., Wang, M., & Tian, Q (2017). Coherent Semantic-Visual Indexing for Large-Scale Image Retrieval in the Cloud. *IEEE Transactions on Image Processing, 26*(9), 4128–4138.
Hong, R., Yang, Y., Wang, M., & Hua, X. (2015). Learning Visual Semantic Relationships for Efficient Visual Retrieval. *IEEE Transactions on Big Data, 1*(4), 152–161.
Kabbur, S., Ning, X., & Karypis, G. (2013). *FISM: Factored item similarity models for top-n recommender systems. SIGKDD*659–667.
Koren, Y., Bell, R. M., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *IEEE Computer,* 30–37.
Li, Y., Tarlow, D., Brockschmidt, M., & Zemel, R. S. (2016). *Gated graph sequence neural networks. ICLR*.
Liu, M., Nie, L., Wang, X., Tian, Q., & Chen, B. (2019). Online data organizer: Micro-video categorization by structure-guided multimodal dictionary learning. *IEEE Transactions on Image Processing, 28*(3), 1235–1247.
Nie, L., Wang, X., Zhang, J., He, X., Zhang, H., Hong, R., & Tian, Q. (2017). *Enhancing micro-video understanding by harnessing external sounds. ACM MM*1192–1200.
Niepert, M., Ahmed, M., & Kutzkov, K. (2016). *Learning convolutional neural networks for graphs. ICML*2014–2023.
Ning, X., & Karypis, G. (2011). *SLIM: Sparse linear methods for top-n recommender systems. ICDM*497–506.
Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). *Deepwalk: Online learning of social representations. SIGKDD*701–710.
Ren, Z., Liang, S., Li, P., Wang, S., & de Rijke, M. (2017). *Social collaborative viewpoint regression with explainable recommendations. WSDM*485–494.
Rendle, S. (2010). *Factorization machines. ICDM*995–1000.
Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). *Bpr: Bayesian personalized ranking from implicit feedback. UAI*452–461.
Shang, X., Di, D., Xiao, J., Cao, Y., Yang, X., & Chua, T. (2019). *Annotating objects and relations in user-generated videos. ICMR*279–287.
Song, W., Xiao, Z., Wang, Y., Charlin, L., Zhang, M., & Tang, J. (2019). *Session-based social recommendation via dynamic graph attention networks. WSDM*555–563.
Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017). *Graph attention networks*.
Wang, H., Zhao, M., Xie, X., Li, W., & Guo, M. (2019). *Knowledge graph convolutional networks for recommender systems. WWW*3307–3313.
Wang, M., Hong, R., Li, G., Zha, Z., Yan, S., & Chua, T. (2012). Event driven web video summarization by tag localization and key-shot identification. *IEEE Transactions on Multimedia, 14*(4), 975–985.
Wang, X., He, X., Cao, Y., Liu, M., & Chua, T. (2019). *KGAT: Knowledge graph attention network for recommendation. KDD*950–958.
Wang, X., He, X., Feng, F., Nie, L., & Chua, T. (2018). *TEM: Tree-enhanced embedding model for explainable recommendation. WWW*1543–1552.
Wang, X., He, X., Nie, L., & Chua, T. (2017). *Item silk road: Recommending items from information domains to social users. SIGIR*185–194.
Wang, X., He, X., Wang, M., Feng, F., & Chua, T. (2019). *Neural graph collaborative filtering. SIGIR*165–174.
Wang, X., Jin, H., Zhang, A., He, X., Xu, T., & Chua, T. (2020). Disentangled Graph Collaborative Filtering. *SIGIR*.
Wang, X., Wang, D., Xu, C., He, X., Cao, Y., & Chua, T. (Wang, Xu, He, Cao, Chua, 2019d). *Explainable reasoning over knowledge graphs for recommendation. AAAI*.
Wang, X., Xu, Y., He, X., Cao, Y., Wang, M., & Chua, T. (2020). Reinforced Negative Sampling over Knowledge Graph for Recommendation. *WWW,* 99–109.
Wei, Y., Wang, X., Nie, L., He, X., Hong, R., & Chua, T. (2019). *MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. ACM MM*1437–1445.
Wong, Y., Chen, S., Mau, S., Sanderson, C., & Lovell, B. C. (2011). *Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. CVPR*74–81.
Wu, L., Sun, P., Fu, Y., Hong, R., Wang, X., & Wang, M. (2019). *A neural influence diffusion model for social recommendation. SIGIR*235–244.
Wu, Q., Zhang, H., Gao, X., He, P., Weng, P., Gao, H., & Chen, G. (2019). *Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. WWW*2091–2102.
Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., & Tan, T. (Tang, Zhu, Wang, Xie, Tan, 2019c). *Session-based recommendation with graph neural networks. AAAI*346–353.
Xiao, J., Ye, H., He, X., Zhang, H., Wu, F., & Chua, T. (2017). *Attentional factorization machines: Learning the weight of feature interactions via attention networks. IJCAI*3119–3125.
Zhang, H., Shen, F., Liu, W., He, X., Luan, H., & Chua, T. (2016). *Discrete collaborative filtering. SIGIR*325–334.
Zheng, L., Lu, C., Jiang, F., Zhang, J., & Yu, P. S. (2018). *Spectral collaborative filtering. RecSys*311–319.
Zheng, Y., Gao, C., He, X., Li, Y., & Jin, D. (2020). Price-aware recommendation with graph convolutional networks. *CoRR, abs/2003.03975*.