Information Sciences 580 (2021) 495–509

Contents lists available at ScienceDirect

Information Sciences

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

Multi-view group representation learning for location-aware group recommendation

Ziyu Lyu [a], Min Yang [a,*], Hui Li [b]

[a] Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, China
[b] School of Informatics, Xiamen University, China

ARTICLE INFO

Article history:
Received 31 July 2019
Received in revised form 10 June 2021
Accepted 24 August 2021
Available online 25 August 2021

Keywords:
Multi-view learning
Group recommendation
Location-aware recommendation

ABSTRACT

With the development of location-based services (LBS), many location-based social sites like Foursquare and Plancast have emerged. People can organize and participate in group activities on those sites. Therefore, recommending venues for group activities is of practical value. However, the group decision making process is complicated, requiring trade-offs among group members. And the data sparsity and cold-start problems make it difficult to make effective group recommendation. In this manuscript, we propose a Multi-view Group Representation Learning (MGPL) framework for location-aware group recommendation. The proposed multi-view group representation learning framework can leverage multiple types of information for deep representation learning of group preferences and incorporate the spatial attributes of locations to further capture the group mobility preferences. Experiments on two real datasets Foursqaure and Plancast show that our method significantly outperforms the-state-of-art approaches.

1. Introduction

Recommender system plays an important role in addressing information expansion and has had a large number of practical applications [1,23]. In recent decades, some event-based social networks such as Plancast[1] and Mafengwo[2] have emerged, which calls for the recommendation for a group of users, namely *group recommendation*. Different from traditional personalized recommendation, group recommendation is a challenging task because the group decision making process requires trade-offs among members' preferences. In addition, the data sparsity and cold-start problems are very severe in group recommendation, which makes it difficult to make effective recommendations. With the development of location-based social services, the inclusion of the location factor in recommender system has brought a new research topic called location-aware recommendation [11,22]. Recently, location-aware recommendation problems have been studied in several scenarios [27,28,37]. Previous studies [44,31,14] have observed that the spatial properties of groups or items have important effects on the selection of group activity venue.

Early studies on group recommendation extend memory-based collaborative filtering (CF) based on aggregation methods [2], including *preference aggregation* and *score aggregation*. But the early aggregation methods are shown to be inadequate with some static aggregation strategies such as average and least-misery (LM) because they ignore to differentiate personal

* Corresponding author.
E-mail address: min.yang@siat.ac.cn (M. Yang).

1 http://plancast.com
2 http://www.mafengwo.cn/

https://doi.org/10.1016/j.ins.2021.08.086
0020-0255/

impacts from different group members. Recently, several model-based group recommendation methods have been proposed for group recommendation [26,43,7,27,18,42]. For example, Liu et al. [26] proposed a personal impact topic model (PIT) for group recommendation and an extended model with social influences. Yuan et al. [43] proposed a generative consensus model for group recommendation. They considered topic-dependent personal impacts, and combined both the group's topic preferences and the group member's personal preferences. Cao et al. [7] utilized the attention network to learn group members' personal impacts on the group decision and proposed the attentive group recommendation method (AGR). Aggregation of group members' personal preferences with the learned influence weights acted as one component of the group preference modelling. Later, the authors incorporated social influences into AGR and proposed a social-enhanced group recommendation method through devising another social based attention network to model social influences from the social followee [8]. HBGG [27] proposed a hierarchical generative geographical model for group activity venue recommendation. HBGG designed a geographical topic model to discover the group mobility behaviors and integrated social connections into one-class collaborative filtering (OCCF) [29] to address the cold-start problem. [18] designed a new model to learn graphical and attentive multi-view embeddings for groups, users and items, from the independent view and the counterpart views within the interaction graph.

Although existing studies have exploited some side information for group recommendation such as social influences [8,13,41] and geographical information [27,43], they have not utilized those complementary information in a comprehensive manner or taken full advantages of available data. The data sparsity and cold-start problems heavily reduce the quality of recommendations. The representation learning of group preference is incomplete, requiring deep understanding and comprehensive representation of group preferences [39,16]. Therefore, it is essential to design a flexible way to incorporate multiple types of information for deep understanding of item characteristics and group preferences, which can bring recommendation performance improvements and further alleviate the cold-start problem. In addition, the spatial-aware representation learning has not been fully explored in previous studies, and most methods ignore the spatial influences during the location-aware group decision making process.

In order to solve the mentioned challenges, we propose a multi-view group presentation learning framework (**MGPL**) for location-aware group recommendation. We investigate group preferences from multi-views for deep understanding of group preferences, by comprehensively exploiting implicit group-item interactions and multiple types of auxiliary information. In addition, the spatial attributes of items (locations) are leveraged in the representation learning of both items and groups to incorporate spatial influences in *location-aware group recommendation*. The main contributions of this manuscript are listed as follows:

- A novel multi-view group representation learning framework is proposed for location-aware group recommendation. The multi-view group representation learning framework allows to depict and learn the group preferences and item properties from different views, by jointly fusing features learned from multi-source side information, and generating more comprehensive representations. The proposed framework can help alleviate the influences of data sparsity and cold-start problem.
- We design a flexible way to incorporate the spatial attributes of items and groups in the representation learning framework for location-aware group recommendation.
- Extensive experiments have been conducted on two real datasets (Foursquare and Plancast). The experiment results show that our proposed method is superior to the state-of-art approaches by a wide margin.

The rest of this manuscript is organized as follows. Section 2 presents related work. Section 3 states the problem definition and demonstrates some background methods. In Section 4, we illustrate the proposed framework. The experimental evaluation is presented in Section 5 and we conclude our paper in Section 6.

2. Related work

2.1. Group recommendation

Early studies on group recommendation are mainly memory-based collaborative filtering approaches by leveraging aggregation methods [3,2] including preference aggregation and score aggregation. The difference between the two is the order of applying the aggregation and recommendation steps. Preference aggregation approaches first aggregate the profiles of all group members and then perform recommendation on the aggregated profile using a classic CF approach (e.g. user-based CF). Score-based aggregation methods firstly compute recommendations for each group member and then aggregate the recommendation scores. Score-based aggregation strategies include the average strategy, the least-misery (minimum) strategy, etc. Aggregation is done by taking for each item the average (averaging) or minimum (least-misery) predicted score per user and considering this aggregated score to be the group's score on the item. The main drawback of aggregation-based methods is that they ignore the influences between users when making the group decision.

In addition, several model-based methods have been proposed for group recommendation. For example, social influences have been explored for group recommendation. Ye and Liu et al. [40] proposed a Social Influence Selection Model (SIS) based on Latent Dirichlet Allocation (LDA) by considering the influences of friendship. They extended SIS to design a Personal

Impact Topic model (PIT) [26], by introducing a personal impact variable to represent the personal influences on the group decision. PIT aggregated the individual preference score with the learned personal impacts. Yuan et al. [43] designed a generative consensus model called COM for group recommendation based on two different assumptions: (i) personal impacts are topic-dependent and (ii) the group decision process depends on the group's topic preferences and the personal preferences of individual group members. Lu et al. [27] proposed a Hierarchical Bayesian Geographical Model (HBGG) for the group activity venue recommendation. Besides considering influences of group membership and topic-aware group preferences, HBGG investigated the geographical influences to model group mobility regions and exploited social connections to develop the social-based collaborative filtering module which can alleviate the data sparsity and cold-start problems.

Recently, deep learning methods have been applied for group recommendation because they demonstrate its effectiveness in recommendation tasks. Cao et al. [7] proposed an attention-based group recommendation model (AGR) where the attention network learned different impacts of each group member on the group decision. Aggregation of each group member's preferences with the learned personal impact weight is represented as one component of the group's preference representation learning. The neural collaborative filtering (NCF) [17] framework has been utilized to model the group-item interactions and predict the group's preference score on each item. Cao et al. [8] extended the attention-based group recommendation model (AGR) through devising another social based attention network to model the social influences from the social followee. He et al. [18] proposed a new method to learn the graphical and attentive multi-view embeddings for groups, users, and items from the independent view and the counterpart views within the interaction graph. Yin et al. [42] proposed a centrality-aware group recommender (CAGR) in which one bipartite graph embedding model, the self-attention mechanism and graph neural networks have been exploited as basic blocks to learn group and user representations in a unified way.

Although existing studies designed some mechanisms to learn the social influences or the multi-view embeddings within the interaction graph, there is no flexible way to comprehensively incorporate multiple types of side information for deep understanding of group preferences during the group decision making process. Multiple types of side information have important effects on recommendation performance, which can help alleviate the data sparsity and cold-start problems. Especially the geographical information has not been considered in recent studies, which is essential for location-aware group recommendation. The data sparsity and cold-start problems are still unsolved.

2.2. Multi-view representation learning

Multi-view representation learning aims to learn the feature representation of multi-view data from different perspectives and obtain comprehensive representations to bring performance improvements. As multi-view data are very prevalent in real data applications, a large number of multi-view representation learning algorithms have been proposed [24,33]. The multi-view representation learning algorithms can be classed into two categories: respectively *multi-view representation alignment* and *multi-view representation fusion*. Multi-view representation alignment methods capture the relationships among multiple views through feature alignment. Multi-view representation fusion combines the features learned from different views into a joint representation.

For the multi-view representation alignment category, some correlation-based models, and distance and similarity-based models have been utilized for feature alignment, such as canonical correlation analysis (CCA) [19] and cross-modal hashing methods [6,45]. Among the multi-view representation fusion methods, the representative examples are graphical models and neural network-based models. Based on the classic Latent Dichilet Allocation (LDA) [5], Chong et al. [9] proposed a multi-modal probabilistic model called multi-class supervised LDA to make effective usage of multi-view data. In addition, several neural network based models have been proposed for multi-view representation fusion due to their effective powers in representation learning. For example, Srivastava and Salakhutdinov [32] proposed a deep multi-modal Restricted Boltzmann Machines (RBM) to learn joint representation of images and texts.

Currently, multi-view representation learning has been utilized to exploit multiple types of information in several recommendation tasks. For example, Collective topic regression (CTR) [36] is an early method leveraging both the rating data and the content information to improve the personalized recommendation performances. Elkahky et al. [12] proposed a multi-view deep learning model to map users and items in a coordinated space and jointly learn user features and item features from different domains for cross-domain recommendation. Cui et al. [10] proposed a Multi-View Recurrent Neural Network (MV-RNN) model for sequential recommendation by incorporating visual and textual information.

However, the multi-view representation learning has not been extensively applied in group recommendation. The data sparsity and cold-start problems make it difficult to capture and learn group preferences only based on the single-view data. The multi-view representation learning capable of incorporating the multiple types of side information and learning the group preferences from different views, can alleviate the cold-start problem and improve recommendation performance.

3. Preliminaries

In this section, we first formulate our problem and then present some basic models utilized in our framework.

3.1. Problem definition

Formally, we have a set of locations V, a set of users U and a set of historical groups G. Each group has a set of group members $U_i = \{u_{i1}, u_{i2}, \ldots, u_{ik}\}$, in which each user member u_{ik} (ik is the user index) comes from the user set ($u_{ik} \in U$). Each location v is associated with its geographical coordinates $l_v =< Latitude, Longitude >$. The historical group visit records are represented as a binary matrix $\mathbf{X} \in \mathbb{R}^{M \times N}$. $\mathbf{X}_{ij} = 1$ indicates that group g_i visits the location v_j, otherwise the value is 0. M is the number of groups and N is the number of locations. In addition, there might be auxiliary information such as social connections, content information and spatio-temporal information. The auxiliary information is constructed as extra side matrix $\mathbf{X}^*$ which can be leveraged for deep representation learning from multi-views.

Given the set of groups G, the set of users U, the set of venues V, the implicit group-location matrix $\mathbf{X}$ and other side information matrix $\mathbf{X}^*$, recommend to an ad hoc user group $g \subseteq G$ the top-N venues to visit.

3.2. Neural collaborative filtering

The Neural Collaborative filtering (NCF) framework [17] is an end-to-end deep learning method for recommendation. It mainly consists of 3 components, respectively the input layer, the embedding layer and neural collaborative layers. The architecture of the Neural Collaborative filtering (NCF) framework is shown in Fig. 1. The bottom component is the input layer which accepts feature vectors for entities, e.g. the user feature vector and the item feature vector. The feature vectors can be customized to a wide range of modeling, such as content-based and neighbour-based feature vectors. The identity of a user and a item works as the input and is transformed to a binary sparse vector with one-hot encoding. Above the input layer is the embedding layer in which a fully connected layer is commonly used to project the sparse representation into a dense representation vector. The output of the embedding acts as the input of the neural collaborative component where some neural collaborative layers model the user-item interactions and generate the prediction scores $\hat{y}$. With the prediction scores and the target scores, optimizing the objective function can obtain the learning parameters in the NCF.

3.3. Autoencoder

Autoencoder [4] is a feedback neural network for learning representations of the input data. The basic autoencoder is typically implemented as a one-hidden layer neural network that takes a input matrix $\mathbf{X}$ and maps it to a hidden representation $\mathbf{z} \in R^K$ with a mapping function as follows:

$$\mathbf{z} = h(\mathbf{X}) = \sigma(\mathbf{W}_a^T\mathbf{X} + \mathbf{b}_a) \tag{1}$$

where $\mathbf{W}_a$ is the weight matrix and $\mathbf{b}_b$ is an offset vector. σ denotes the mapping function. The hidden representation is reconstructed as $\hat{\mathbf{X}}$ through the following formula:

$$\hat{\mathbf{X}} = \sigma(\mathbf{W}_b^T\mathbf{X} + \mathbf{b}_b) \tag{2}$$

$\mathbf{W}_b$ is the weight matrix and $\mathbf{b}_b$ is an offset vector. The autoencoder is trained by minimizing the average reconstruction error:

$$\mathop{\arg\min}_{\mathbf{W}_a,\mathbf{W}_b,\mathbf{b}_a,\mathbf{b}_b} \frac{1}{n}\sum_{i=1}^{n} l(\mathbf{X}_i, \hat{\mathbf{X}}_i)$$

l is the loss function such as the square loss or the cross entropy loss. $\mathbf{W}_a, \mathbf{W}_b, \mathbf{b}_a, \mathbf{b}_b$are the learning parameters.

The denoising autoencoder (DAE) [35] extends the classical autoencoder by reconstructing each data point $\mathbf{X}$ from its corrupted version $\tilde{\mathbf{X}}$. DAE can discover more robust features with the corruption operation. Corruption is done by randomly making some of the input as zero.

Stacked denoising autoencoder (SDAE) [2,38] is stacking of denoising autoencoders. SDAE has L layers in which the first L/2 layers of the network act as an encoder and the last L/2 layers act as a decoder. The input layer is a corrupted version $\tilde{\mathbf{X}}$ of the clean input data. Fig. 2 shows an example of SDAE with $L = 4$. The last layer outputs the reconstructed presentation. The parameters of all layers $\theta = \{\mathbf{W}_l \mathbf{b}_l\}, l = 1 \ldots L$ are learned by minimizing the loss function as follows:

$$\min_\theta ||\mathbf{X} - \mathbf{X}_L||^2 + \lambda \sum_l ||\mathbf{W}_l||^2 \tag{3}$$

λ is the regularization parameter.

4. Method

We proposed a multi-view group representation learning (MGPL) framework for location-aware group recommendation. For deep understanding of group preferences in the complicated process of group decision making, we model both the group preferences and the item representation from multi-views by combining the features from multi-view data into joint rep-

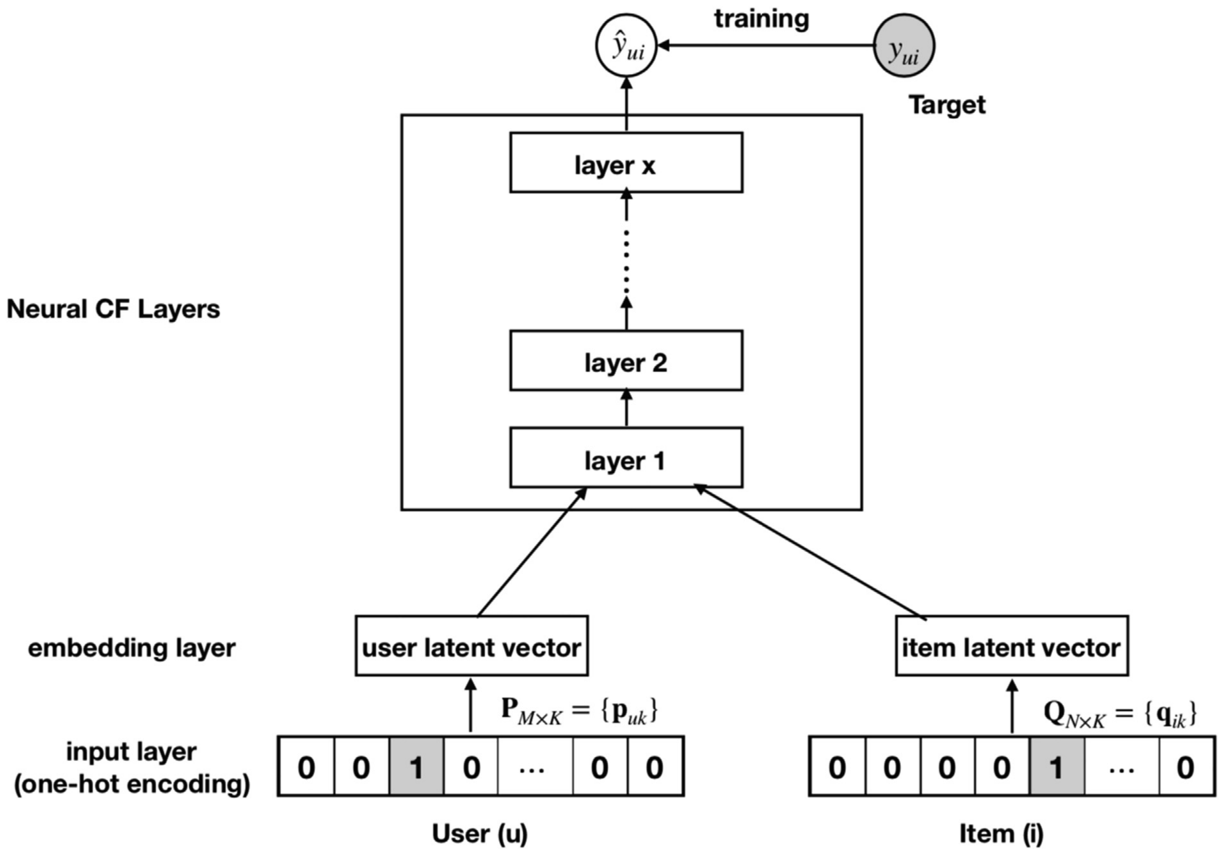


Fig. 1. Neural Collaborative Filtering Framework.

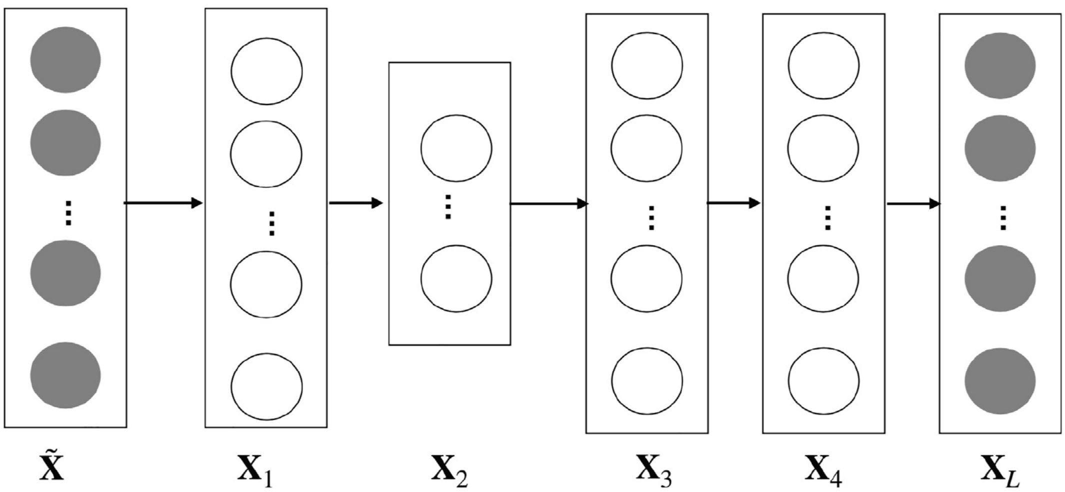


Fig. 2. Example of Stacked Denoising Autoencoder (SDAE) with L = 4.

resentation. The deep representation learning can alleviate the influences due to data sparisty and deal with the cold-start problem.

Our multi-view group representation learning framework mainly contains three parts, respectively location-aware item representation, multi-view group preference modeling and group-item interaction prediction. Fig. 3 shows the architecture of the multi-view representation learning framework. The details of the three parts are illustrated in the following sections.

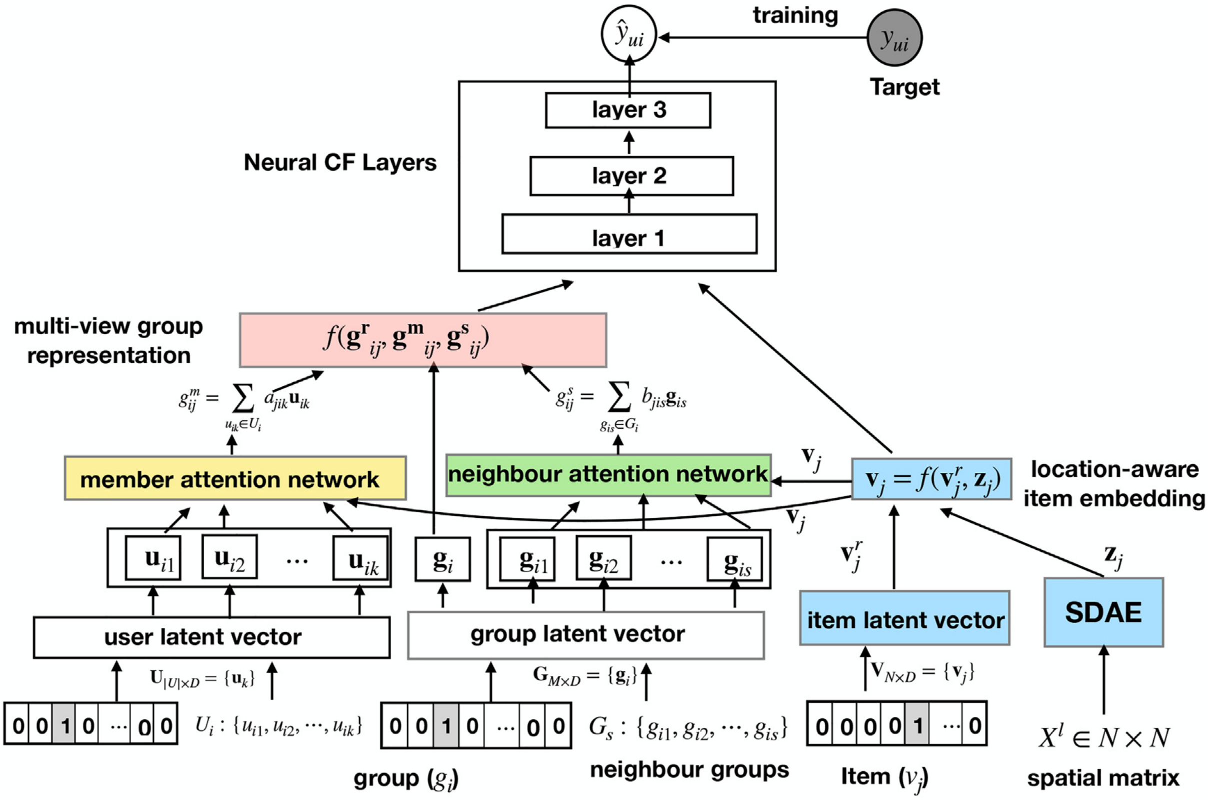


Fig. 3. The architecture of the multi-view representation learning framework.

4.1. Location-aware item representation

The spatial attributes have played an important role in location-aware recommendation. Therefore, we design a location-aware item representation learning module for the location-ware group recommendation, by exploiting spatial attributes of items in the item representation learning. The bottom-right part (blue color) in Fig. 3 demonstrates the location-aware item representation.

Firstly, each item v_j is represented with one-hot encoding and a fully-connected layer is adopted to map the sparse one-hot representation into a dense item representation $\mathbf{v}^r$. In order to exploit spatial attributes in the item representation learning, a stacked denoising autoencoder (SDAE) with 4 layers is utilized to encode the spatial attributes. By considering the geographical distances between two items, we compute the pair-wise item spatial similarity and construct the auxiliary spatial matrix $\mathbf{X}^l \in \mathbb{R}^{N\times N}$. N is the number of items and each entry x_{ij} in the spatial matrix measures the spatial influences[3] between the item v_i and item v_j. We compute x_{ij} based on Eq. (4). l_i and l_j are the geographical coordinates for the location v_i and v_j.

$$x_{ij} = 1 - \frac{\text{geodist}(l_i, l_j)}{\max_{v_i \in V} \text{geodist}(l_i, l_j)} \tag{4}$$

The i_{th} row of the formed spatial matrix $\mathbf{X}^l_i$ can be seen as the spatial feature vector for the item v_j. In order to obtain the spatial embedding of items, the spatial matrix is corrupted by dropping-out noises and the corrupted version $\tilde{\mathbf{X}}^l$ acts as the input of the stacked denoising autoencoder (SDAE) with 4 Layers. The first 2 layers of SDAE acts as an encoder to generate the hidden representation $\mathbf{z}$ and the last 2 layers reconstruct from the hidden representation to the raw spatial matrix $\mathbf{X}^l$. The hidden representation $\mathbf{z}_j$ for item v_j is seen as the item's spatial embedding.

To obtain the location-aware item representation, the latent item vector and the spatial embedding from SDAE are combined to generate the location-aware item representation as in Eq. (5):

$$\mathbf{v} = f(\mathbf{v}^r, \mathbf{z}) \tag{5}$$

[3] There might be many methods to define the spatial influences. Here we use the simple formula to measure the spatial influences based on the pairwise geographical distance. Exploration of other complex definitions is beyond our main focus.

f is the fusion function to integrate the feature representation from multi-views. As the feature-level fusion, some classical approaches can be utilized, such as feature augmentation through feature addition or feature combination by concatenating the multi-view features vectors. In this manuscript, we augment the item latent feature vector $\mathbf{v}^r$ with the spatial item embedding $\mathbf{z}$ to obtain the location-aware item representation $\mathbf{v}$ as follows:

$$\mathbf{v} = \mathbf{v}^r + \mathbf{z} \tag{6}$$

4.2. Group preference modeling

Different from personalized recommendation for individuals, the main challenge of the group recommendation task is how to model and represent group preferences because it requires trade-offs among different group members' preferences. It is difficult to accurately represent the group preferences due to data sparsity. And the cold-start problem is very severe as most of groups are ad hoc groups which form occasionally. For deep understanding of group preferences and tackling the cold-start problem, we learn and represent the group preferences from multi-views by leveraging several types of complementary information.

4.2.1. Latent group preference representation

For the implicit check-in records, we can learn the group latent factor. Firstly, each group is represented as a binarized sparse vector with one-hot encoding and an embedding layer (commonly a fully connected layer) is adopted to transform the sparse vector into a dense vector representation as the group latent preference representation $\mathbf{g}^r$. Namely, for a group g_i, its latent preference representation is $\mathbf{g}_i^r \in R^D$. D is the embedding size of the latent dense representation.

4.2.2. User aggregation group preference presentation

Aggregating individual members' preferences has been extensively explored in existing studies [2,3], which has been the main component of modeling the group preferences. Several aggregation strategies has been adopted, e.g. average, least-misery, etc. Recently, the attention network [34] has been adopted to learn the dynamic aggregation weights to aggregate user members' preferences. We also utilize the attention mechanism to learn the *user aggregation group preferences* as one component of the group preference representation.

Similar with [7], a member attention network (the left part in yellow color in Fig. 3) is adopted to learn the dynamic aggregation weights for aggregating user member preferences. For a group g_i, it contains a set of group members $U_i = \{u_{i1}, u_{i2}, \ldots, u_{ik}\}$ and ik is the user index. The member attention network learns the dynamic aggregation weights a_{jik} for aggregating user member preferences. a_{jik} denotes the impact of the user member u_{ik} when the group g_i makes the group decision on the location v_j. The user impact weights are relevant with the group members' personal preferences and the item's characteristics. Through the member attention network, a_{jik} is computed as in Eq. (7).

$$O_{jik} = \mathbf{h}^T ReLU(\mathbf{W}_v \mathbf{v}_j + \mathbf{W}_u \mathbf{u}_{ik} + \mathbf{b}) \tag{7}$$

$$a_{jik} = softmax(O_{jik}) = \frac{\exp(O_{jik})}{\sum_{u_{ik} \in U_i} \exp(O_{jik})}$$

where $\mathbf{W}_v$ and $\mathbf{W}_u$ are weight matrices of the attention network that convert the user member embedding and the location-aware item embedding into the hidden layer. $\mathbf{h}^T$ is the hidden vector, and $\mathbf{b}$ is the bias vector of the hidden layer. ReLU is used as the activation function of the hidden layer and outputs the user member influence score O_{jik}. Softmax is adopted to normalize the influence score and makes it as a probabilistic interpretation. Through the member attention network, the learned user member weight a_{jik} indicates the influences from the user member u_{ik} on the group decision for the item v_j, differentiating influences of different user members. As the location-aware item embedding $\mathbf{v}$ is exploited in the attention network, the learning aggregation weights also encode the item's spatial influences. The *user aggregation group preference* representation is constructed by aggregating user member's preferences $\mathbf{u}$ with the learned weights $\mathbf{a}$ as in Eq. (8).

$$\mathbf{g}_{ij}^m = \sum_{u_{ik} \in U_i} a_{jik} \mathbf{u}_{ik} \tag{8}$$

$\mathbf{u}_{ik}$ is the group member's user latent factor for u_{ik} from the user embedding layer. a_{jik} denotes the influence of the user member u_{ik} when the group g_i selected the location v_j. Specifically, the member attention network accepts the individual user preference of $u_{i1}, u_{i2}, \ldots, u_{ik}$ as input and outputs the user aggregation group preference $\mathbf{g}_{ij}^m$, as shown in the yellow part in Fig. 3.

4.2.3. Neighbour group preference presentation

As most groups are ad hoc groups, only few groups have many historical records, which causes the data sparsity and cold-start problems. In order to alleviate those effects due to data sparsity and cold-start problems, we propose to learn a collective group representation by leveraging the **neighbour groups' preferences**.

For a group g_i, we can find a set of k-nearest neighbour groups G_i^s ($G_i^s \subset G$) by using some similar metrics e.g. Jaccard similarity and cosine similarity. When obtaining the neighbour groups, the neighbour group preference representation is constructed by weighty aggregating each neighbour groups' preference in the neighbour groups. A neighbour attention network (the middle part in green color in Fig. 3) is designed to learn the different influences of different neighbour groups. Specifically, for a group g_i and its neighbour groups $G_i^s = \{g_{i1}, g_{i2}, \dots, g_{is}\}$ (is the group index), the influence weight b_{jis} from the neighbour group g_{is} on the item v_j is computed through the neighbour attention network as in Eq. (9).

$$O_{jis} = \mathbf{h}^T ReLU(\mathbf{W}_l \mathbf{v}_j + \mathbf{W}_s \mathbf{g}_{is} + \mathbf{b}_s) \tag{9}$$

$$b_{jis} = softmax(O_{jis}) = \frac{\exp(O_{jis})}{\sum_{g_{is} \in g_i^s} \exp(O_{jis})}$$

where $\mathbf{W}_l$ and $\mathbf{W}_s$ are weight matrices of the neighbour attention network. $\mathbf{h}^T$ is the hidden vector, and $\mathbf{b}_{jis}$ is the bias vector of the hidden layer. $\mathbf{g}_s$ is the group latent vector of the group g_{is} in the neighbour groups G_i^s. With the learned neighbour influence weights $\mathbf{b}_{jis}$, the *neighbour group preference* $\mathbf{g}_{ij}^s$ is generated by aggregating the neighbour groups' latent preference vectors as defined in Eq. (10).

$$\mathbf{g}_{ij}^s = \sum_{g_{is} \in G_i^s} b_{jis} \mathbf{g}_{is} \tag{10}$$

4.2.4. Multi-view group preference

The multi-view group preference representation is constructed by fusing the above different types of group preference representation. Through a fusion function f_m, the multi-view group preference is performed as follows (red color in Fig. 3):

$$\mathbf{g}_{ij} = f_m\left(\mathbf{g}_i^r, \mathbf{g}_{ij}^m, \mathbf{g}_{ij}^s\right) \tag{11}$$

The fusion function f_m can be some simple operations,e.g. element-wise vector addition $\mathbf{g}_{ij} = (\mathbf{g}_i^r + \mathbf{g}_{ij}^m + \mathbf{g}_{ij}^s)$ or concatenation $\mathbf{g}_{ij}^T = \left[\mathbf{g}_i^{rT}, \mathbf{g}_{ij}^{mT}, \mathbf{g}_{ij}^{sT}\right]^T$. In addition, some complex models can be adopted to integrate the multi-view group representation, such as neural network based models. In this manuscript, we use the concatenation operation for multi-view feature fusion. We leave the exploration of more complex operations in future work. The three types of group preference presentation including the group latent preference $\mathbf{g}_i^r$, the user aggregation group preference $\mathbf{g}_{ij}^m$ and the neighbour group preference $\mathbf{g}_{ij}^s$, are fused to obtain the multi-view group preference representation $\mathbf{g}_{ij}^T = \left[\mathbf{g}_i^{rT}, \mathbf{g}_{ij}^{mT}, \mathbf{g}_{ij}^{sT}\right]^T$.

4.3. Group-item interaction

With the multi-view representation of items and groups, the next step is to design the group-item interaction functions to perform the prediction. As mentioned in previous work [17], NCF is more generalizable than the traditional matrix factorization (MF) model and MF models can be generalized from NCF. Therefore, we adopt neural collaborative layers like NCF to perform the prediction. We design three neural collaborative layers to act as the interaction functions and perform the final group preference prediction. The network structure in the prediction layer follows a tower structure in which the bottom layer is the widest and each successive layer has a smaller number of neurons. The output of one neural collaborative layer acts as the input of the successive neural collaborative layer. The input of the first neural collaborative layer is the concatenation of the location-aware item embedding $\mathbf{v}$ and the multi-view group preference representation $\mathbf{g}$. In each neural collaborative layer, the hidden vector $\mathbf{h}_l$ is generated as in Eq. (12).

$$\mathbf{h}_l = \phi(\mathbf{W}_l^T \mathbf{h}_{l-1} + \mathbf{b}_l) \tag{12}$$

$\mathbf{W}_l$ and $\mathbf{b}_l$ are the weight matrix and bias vector at the l_{th} layer. ϕ_l is the activation function used at the l_{th} neural collaborative layer. ReLU is used as the activation function as it shows better performance. In the last neural collaborative layer, the hidden vector $\mathbf{h}_l$ is mapped to the prediction score as follows:

$$\hat{y}_{ij} = \sigma(\mathbf{W}^T \mathbf{h}_l) \tag{13}$$

$\mathbf{W}$ is the weight matrix and σ is the sigmoid function. The input of the first neural collaborative layer is the concatenation of the location-aware item embedding $\mathbf{v}$ and the multi-view group preference representation $\mathbf{g}$. The prediction score $\hat{y}_{ij}$ denotes the preference score of the group g_i on the location v_j.

4.4. Model optimization

As the proposed method is an end-to-end method, the representation learning and the prediction part are jointly learning. The objective is to minimize the overall loss function as defined in Eq. (14).

$$L = (1-\lambda_a)L_r + \lambda_a L_a + \lambda_g \sum_i ||\mathbf{g}_i||^2 + \lambda_v \sum_j ||\mathbf{v}_j||^2 + \lambda_u \sum_k ||\mathbf{u}_k||^2 \tag{14}$$

It mainly consists of two parts. The first term L_r is the prediction loss and the second term L_a is the loss of the stacked denoising autoencoder (SDAE) for location-aware item embedding. The parameter λ_a is to control the weight of SDAE. The remaining terms are the regularization terms. $\lambda_g, \lambda_v, \lambda_u$ are the regularization coefficients.

As we work on the implicit data and our task is Top-N recommendation, we adopt the weighted point-wise loss function as the prediction loss as defined in Eq. (15). $\hat{y}_{ij}$ is the predicted preference score of the group g_i for the item v_j as obtained in Eq. (13). w_{ij} is the confidence weight for each training instance and measures the confidence of the true rating y. A plausible choice for w_{ij} would be $w_{ij} = 1 + \alpha y_{ij}$.

$$L_r = \sum_{ij} w_{ij}(y_{ij} - \hat{y}_{ij})^2 \tag{15}$$

The second term L_a denotes the loss of the stacked denoising autoencoder (SDAE) for location-aware item embedding.

$$L_a = \sum_{l=1}^{L} \sum_j ||\mathbf{X}^l_{lj} - \hat{\mathbf{X}}^l_{lj}||^2 \tag{16}$$

The optimization is performed by using the Adam optimization method [20] in deep learning infrastructures. After the training process, we can obtain the learned latent representation $\{\mathbf{g}, \mathbf{v}, \mathbf{u}\}$, attention weights $\{\mathbf{a}, \mathbf{b}\}$ and model parameters $\{\mathbf{W}_*, \mathbf{b}_*\}$.

Given a test group g_t (an existing group or a new group), our proposed method can predict the group's preference score $\hat{y}_{tj}$ for each location v_j with the learned parameters. The top-N locations ranked on the predicted preference scores are the recommendation results. For a new group or a cold-start group, our method is able to model the group preferences by leveraging the **user aggregation group preference** and the **neighbour group preference**, and generate the recommendation results.

5. Experiment

In this section, we conduct extensive experiments on two public datasets to answer the following four research questions.

- **RQ1:** How does our proposed method MGPL perform, in comparison with state-of-the-art methods?
- **RQ2:** How do the designed different components (i.e. the neighbour group representation and the location-aware embedding) contribute to the performance of our proposed method?
- **RQ3:** How is the effectiveness of our method to overcome the cold-start problem?
- **RQ4**: How does the hyper-parameter (e.g. the control weight λ_a) affect the performance?

5.1. Dataset

Two real datasets are used for experimental evaluation, respectively Plancast [25] and Foursquare [15]. Plancast is an event-based social network where an event consists of the event description, a group of participants, and the event venue. Each event venue has its geographical coordinates. In Plancast, an event is treated as a group decision, the event participants are seen as the group members and the event venue is the group's selected venue. Foursquare is a location-based social network, where users can record their footprints. As there is no explicit group information in Foursquare, we regard a set of friends who check-in at the same venue within one hour time-difference as a group check-in, similar to previous work [30,27]. Then, the set of friends become the group members and the checked-in venue is their selected location. For example, assume that three users u_1, u_2, u_3 are mutual friends (i.e., they form a clique in the social graph) and they checked-in a venue v within a 1-h time difference. Then, a group event is formed in which u_1, u_2, u_3 are the group members and v is the venue.

After forming the group events, we preprocess the two datasets by removing groups who have only a single member (since these correspond to venue selections by individuals), and locations which have been visited only once, similar with previous work [43]. The statistics for both datasets are shown in Table 1, including the number of groups, the number of users and the number of venues, after the preprocessing process. There are 28077 groups, which come from a total of 38184 users, and 8574 venues in Plancast. In Foursquare, there are 6008 groups, 4150 users, and 952 venues. The average group size in Plancast is 9.08 and the average group size for Foursquare is 2.12. Compared to location-based social networks such as Foursquare, event-based social networks tend to have larger groups. And the average number of visited venues for a group in Plancast and Foursquare is very small, respectively 1.06 for Plancast and 1.24 for Foursquare. The numbers indicate

Table 1
Statistics (after preprocessing).

Dataset	Plancast	Foursquare
# of groups $\lvert G \rvert$	28077	6008
# of users $\lvert U \rvert$	38184	4150
# of venues $\lvert V \rvert$	8574	952
average group size	9.08	2.12
average # of a group's check-in venues	1.06	1.24

that existing groups have fewer check-in records and most groups are cold-start groups (Cold-start groups are groups which have no or few records in training set). For both datasets, 15% of group events have been randomly marked off as the test set and 5% of group events have been used for learning the optimal values of the model parameters. The remaining 80% of the events are used as training data.

5.2. Experiment metrics

Two popular metrics are used to evaluate the quality of our model and its competitors, namely Precision@N and Recall@N [26,43,2]. Precision@N is the number of correctly predicted locations divided by the total number N of recommendations made. Recall@N is the ratio of recovered group events in the test set. Let R_g be the set of top-N recommendation items and T_g is the test data for group g. Let $\mathcal{G}_{test}$ be the set of test groups. We denote by $H_g = \lvert R_g \cap T_g \rvert$ the number of correctly predicted locations with regard to group g (i.e., the number of successfully predicted locations). Then, Precision@N and Recall@N are defined as:

$$Precision@N = \frac{\sum_{g \in \mathcal{G}_{test}} H_g}{N \cdot \lvert \mathcal{G}_{test} \rvert}, \quad Recall@N = \frac{\sum_{g \in \mathcal{G}_{test}} H_g}{\sum_{g \in \mathcal{G}_{test}} T_g}. \tag{17}$$

5.3. Compared methods

We compare our proposed method Multi-view Group Representation Learning model (MGPL) with the-state-of-art methods. The first class of competitors are score-based aggregation methods. For this class, the user-based collaborative filtering method [1] is used to estimate individual preferences. Three representative approaches in this class are used as compared methods: respectively AVE-CF, LM-CF and RD-CF [2]. The second class of competitors are some advanced models for group recommendation, including COM [43], PIT [26], HBGG [27], AGR [7], SoAGREE [8], GAME [18], and CAGR [42]. COM and PIT are probabilistic topic models. HBGG is a hierarchical Bayesian Geographical model which models the group mobility regions for location-ware group recommendation. AGR, SoAGREE, GAME, and CAGR are deep learning methods for group recommendation.

All compared methods are described in details as follows:

- AVE-CF [2]: User-based CF is used to estimate the preference scores of items for each user. Given a user group, the recommendation score for an item is derived by averaging the item's preference scores by all group members.
- LM-CF [2]: Same as AVE-CF, but the recommendation score for an item is the smallest score given to the item by any group member (least misery).
- RD-CF [2]: This is the method proposed in [2]. The group recommendation score is estimated by combining *relevance* and *disagreement* in the group. Relevance of the item to the group is based on AVE-CF, while disagreement is the average pairwise difference of recommendation scores by group members.
- PIT [26]: This approach uses an Author-Topic model to learn individual preferences. Also, it models personal impacts during the group decision and uses them to aggregate individual preferences.
- COM [43]: This method combines both topic-dependent group preferences and user personal preferences by using the learned topic-dependent personal impacts within a specific group.
- AGR [7]: AGR is based on the neural collaborative filtering framework. The authors designed an attention network to learn the dynamic aggregation weights of different group members and aggregated the group members' preferences as one component of the group preference representation.
- HBGG [27]: HBGG is one Hierarchical Bayesian Geographical model, which models group mobility regions and integrate social influences into one-class collaborative filtering for location-aware group recommendation.
- SoAGREE [8]: This method is an extension of AGR, by developing another social based attention mechanism to incorporate social influences from the social followee.
- GAME [18]: GAME proposes a method to learn the multi-view embeddings of users, items and groups from both the individual view and the counterpart views.

- GAGR [42]: This method proposes a centrality-aware group recommendation method, by combining one bipartite graph embedding model, the attention mechanism, and graph convolutional networks.
- MGPL: This is our proposed multi-view group representation learning method for location-ware group recommendation in Section 4. We compute the pair-wise group cosine similarity (between two group latent vectors) and collect the top-30 similar groups as neighbour groups in Section 4.2.3.

We used 5% of group events to learn the optimal parameter settings and report the results with the optimal setting in Section 5.4. For PIT, the optimal setting is $\alpha = 50/K, \beta = 0.01$ and $\gamma = 0.01$. For COM, we get the following optimal hyperparameter values: $\alpha = 50/K, \beta = \eta = \rho = 0.01$. For HBGG, we have the optimal values $\lambda_g = 1, \lambda_u = 0.01, \lambda_v = 0.01$ and $\lambda_s = 0.01$ for social-based collaborative filtering. And we get $\alpha = 50/K, \eta = 10/R, \beta = \gamma = \omega = 0.01$ for the group geographical topic model. The optimal λ value for controlling the group geographical topic model and social-based collaborative filtering is $\lambda = 0.8$ for Plancast and $\lambda = 0.7$ for Foursquare. For AGR and SoAGREE, the dimension of the embedding layer and the first hidden layer are set as 32 for both datasets. For GAME, the embedding size is 32 for both datasets. For CAGR, the embedding size is 64 for both datasets. For our proposed method MGPL, the optimal size of the embedding layer, the hidden representation of SDAE and the first hidden layer of the first neural collaborative filtering layer are set as 50. The confidence weight parameter α is set as the optimal value 0.2. The regularized coefficients are $\lambda_g = 0.01, \lambda_v = 0.01, \lambda_u = 0.01$. The optimal value λ_a for controlling the weight of the stacked denoising autoencoder loss is 0.7 for both datasets. And we apply Adam optimizer [21] for all models, where the batch size, learning rate for deep learning methods, are searched in $[32, 64, 128, 256], [0.001, 0.005, 0.01, 0.05, 0.1]$.

5.4. Experimental results

5.4.1. Overall performance comparison (RQ1)

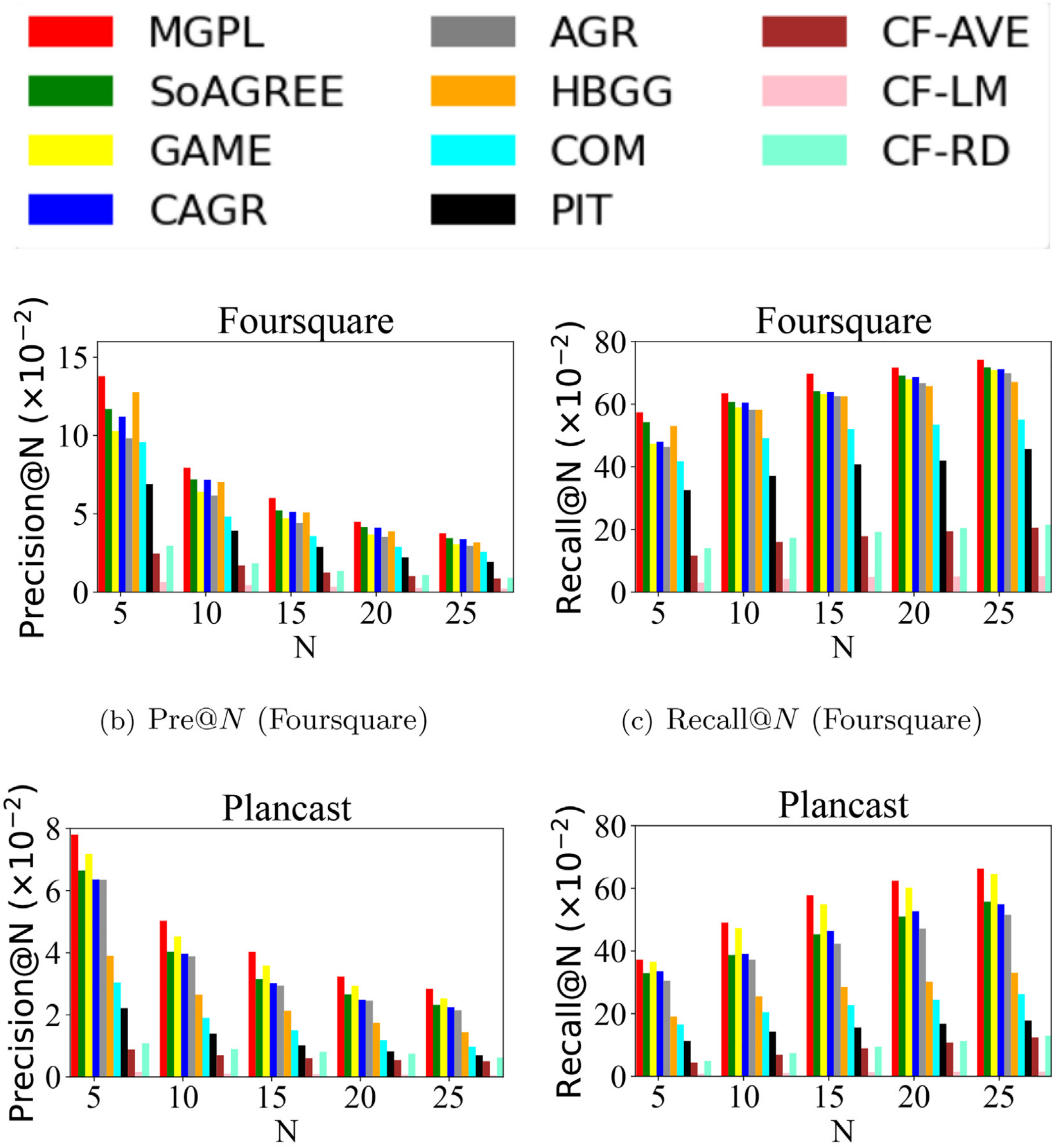
This section demonstrates the performance of our method MGPL, compared with the state-of-the-art methods. The performance comparison results of our proposed method with the state-of-the-art methods at Precision@N and Recall@N for Foursquare and Plancast are shown in Fig. 4. For both datasets, the low precision and recall values are due to data sparsity. It has been noted in previous work [27,7,43] that recommendation on sparse datasets has relatively low precision and recall values. The values we report here are consistent with this observation. We focus on comparing the relative performance of the compared methods and the improvements of our method.

For the model-based methods, we report the best performances with the optimal parameters. Generally, model-based methods perform better than memory-based methods. Our method MGPL outperforms all the compared methods for all metrics on both datasets. In Foursquare, SoAGREE is the best competitor for all Precision and Recall metrics, expect Precision@5. The best competitor at Precision@5 is HBGG. The improvement at Precision@5 over the best competitor (HBGG) is 7.98%, and the improvement at Recall@5 over the best competitor (SoAGREE) is 5.74%. The largest improvements are 15.17% at Precision@15 and 8.76% at Recall@15. In Plancast, the best competitor for all metrics is GAME. The improvements over the best competitor GAME are respectively 8.11% in Precision@5 and 6.33% in Recall@5. The largest improvements are 11.69 % in Precision@15 and 11.21% in Recall@15. The results indicate that our multi-view representation learning framework incorporating multi-view data for representation learning of group preferences and item properties actually improves recommendation performance. To verify the significance of the performance improvements, the paired t-test is conducted. The test result ($p < 0.05$) verify that the improvements are statistically significant.

5.4.2. Ablation studies (RQ2)

The superior performance of our proposed multi-view group presentation learning framework indicates the effectiveness of the end-to-end deep learning based solution for group recommendation. To answer the research question 2, we perform ablation studies to investigate the impacts of the two main components, especially the neighbour group representation and the location-aware embedding. We have two variants of MGPL, respectively MGPL-neighbour and MGPL-geo. MGPL-neighbour denotes the method integrating the neighbour group representation only without the location-aware item embedding. MGPL-geo denotes the method integrating the location-aware item embedding only. The comparison results of MGPL with the two variants are shown in Tables 2 and 3, respectively for Foursquare and Plancast.

We can see that MGPL outperforms the two variants for both datasets. It indicates that the neighbour group representation and the location-aware embedding are both beneficial for location-aware group recommendation. Integration of them into our proposed method can further improve recommendation performance. In addition, we observe that the two variants show different performance on different datasets. In Foursquare, MGPL-geo performs better than MGPL-neighbour. However, MGPL-neighbour outperforms MGPL-geo in Plancast. The differences might be due to the characteristics of the different datasets. Foursquare is a location-based social networks where the spatial attributes play a vital role in user behaviors. The spatial factor has more effects on group behaviors [28,15]. However, Plancast is an event-based social network in which collective groups (neighbour groups) have more effects.



(b) Pre@N (Foursquare) (c) Recall@N (Foursquare)

(d) Pre@N (Plancast) (e) Recall@N (Plancast)

Fig. 4. Precision@N and Recall@N for Foursquare and Plancast.

Table 2
Performance Comparison of Different Variants of MGPL for Foursquare (%).

Method	Precision@5	Precision@10	Recall@5	Recall@10
MGPL	**13.79**	**7.95**	**57.37**	**63.44**
MGPL-neighbour	10.71	6.38	50.89	58.16
MGPL-geo	11.83	6.90	52.68	58.93

5.4.3. Cold-start problem (RQ3)

The cold-start problem is very severe in group recommendation because most of groups are ad hoc groups which allows the numerous possible combinations of users that can potentially form groups. In order to evaluate the ability to alleviate the cold-start problem with the our proposed multi-view representation learning framework, we conduct the experiment to evaluate the recommendation performance on cold-start groups. We define that cold-start groups are groups which have

Table 3
Performance Comparison of Different Variants of MGPL for Plancast (%).

Method	Precision@5	Precision@10	Recall@5	Recall@10
MGPL	**7.80**	**5.03**	**37.23**	**49.02**
MGPL-neighbour	7.25	4.87	36.23	47.52
MGPL-geo	6.76	4.27	34.23	46.02

no or few records in training set. In our datasets, about 90.4% of the groups in the test set in Plancast and about 76.25% of the groups in the test set in Foursquare are cold-start groups.

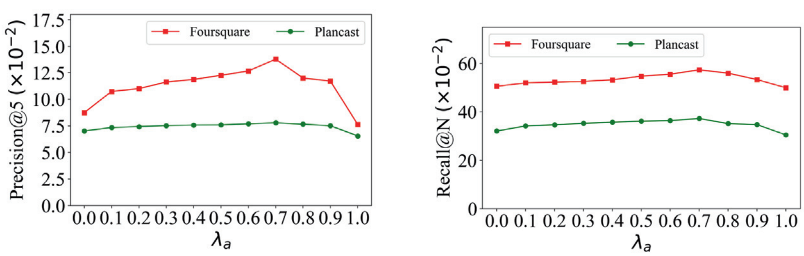
Table 4 represents the Precision@5 and Recall@5 performance of all competitors on cold-start groups for both datasets (%). The bold line indicates the best results and the line with italic font represents the second best methods. In Foursquare, we can see that our proposed method MGPL significantly outperforms all compared methods. The best competitor is HBGG while the performance of SoAGREE on the cold-start groups drops remarkably. The improvements of our proposed method MGPL over the best competitor HBGG in Foursquare are about 6.29% at *Precision*@5 and 3.35% at *Recall*@5 respectively while the improvements of MGPL over SoAGREE are respectively 27.15% at *Precision*@5 and 23.70% at *Recall*@5. In Plancast, the improvements of our proposed method MGPL over the best competitor GAME are about 10.26% at *Precision*@5 and 9.21% at *Recall*@5 respectively. We observe that our method can have larger improvements on more sparse dataset Plancast. The results indicate that our method MGPL is much more effective in handling cold-start groups in group recommendation, in comparison with the state-of-the-art methods. The multi-view representation learning is beneficial to tackling the cold-start problem because MGPL can have deep understanding and comprehensive representations of group preferences and item characteristics by flexibly integrating multiple complementary information.

5.4.4. Effects of λ_a (RQ4)

The parameter λ_a controls the weight of stacked denoising autoencoder (SDAE) in the optimization process, which might have effects on the recommendation performance. Therefore, we perform experiments to investigate the effects of different λ_a. The variation of λ_a is from 0 to 1, stepping with 0.1. Fig. 5 shows the recommendation performance at *Precision*@5 and *Recall*@5 over different λ_a for the two datasets. We can observe that different λ_a actually has different recommendation performance with the optimal λ_a at 0.7 for both datasets. When the parameter λ_a is 0 or 1, the performance is poor which

Table 4
Recommendation Performance for Cold-Start Groups (%).

Method	Foursquare		Plancast	
	Precision@5	Recall@5	Precision@5	Recall@5
MGPL	**11.66**	**61.96**	**7.63**	**37.01**
GAME	8.75	46.33	*6.92*	*33.89*
CAGR	9.32	50.12	6.24	32.09
SoAGREE	9.17	50.09	6.43	31.47
AGR	8.32	44.25	6.14	28.79
HBGG	*10.97*	*59.95*	3.86	18.86
COM	8.00	45.16	2.90	15.46
PIT	5.91	37.78	2.11	11.62
CF-AVE	1.28	8.21	0.81	4.30
CF-LM	0.23	1.46	0.13	1.31
CF-RD	1.47	9.34	0.88	4.38



(a) Precision@5 (b) Recall@5

Fig. 5. The effects of λ_a at Precision@N and Recall@N for Foursquare and Plancast.

indicates the effectiveness of integrating both the rating loss and the loss of SDAE from the location-aware item embedding in the optimization process.

6. Conclusions

In this manuscript, a multi-view group representation learning framework (MGPL) is proposed for location-aware group recommendation. MGPL learns the group preference representation from multi-views by integrating the group members' aggregation preferences and neighbour groups' preferences. And the spatial attributes are exploited into the item representation learning and generate the location-aware item representation. The experimental evaluation on real datasets demonstrate our method is superior to the state-of-the-art methods. And the effective recommendation performance demonstrates MGPL can alleviate the influences of data sparsity and address the cold-start problem.

Currently, the fusion of multi-view group preferences utilizes the simple feature combination methods like the concatenation of feature vectors from different views. In the future, more complex models like multiple perception layers (MLP) or self-attention network, can be adopted for multi-view feature fusion. And more advanced models will be explored for depicting the group-item interactions. In addition, the temporal dynamics in group activities will be considered in future work.

CRediT authorship contribution statement

Ziyu Lyu: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing - original draft, Writing - review & editing. **Min Yang:** Conceptualization, Funding acquisition, Methodology, Project administration, Supervision, Writing - original draft, Writing - review & editing. **Hui Li:** Conceptualization, Methodology, Validation, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

We would like to thank the reviewers for their invaluable comments. Ziyu Lyu was supported by National Natural Science Foundation of China (No. 62002352). Min Yang was partially supported by the Natural Science Foundation of Guangdong Province of China (No. 2019A1515011705), Youth Innovation Promotion Association of CAS China (No. 2020357), Shenzhen Science and Technology Innovation Program (Grant No. KQTD20190929172835662), Shenzhen Basic Research Foundation (No. JCYJ20200109113441941).

References

[1] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (6) (2005) 734–749.
[2] S. Amer-Yahia, S.B. Roy, A. Chawla, G. Das, C. Yu, Group recommendation: Semantics and efficiency, PVLDB 2 (1) (2009) 754–765.
[3] L. Baltrunas, T. Makcinskas, F. Ricci, Group recommendations with rank aggregation and collaborative filtering. In: Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010, Barcelona, Spain, September 26–30, 2010. pp. 119–126. .
[4] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, Greedy layer-wise training of deep networks. In: Proceedings of the 19th International Conference on Neural Information Processing Systems. NIPS'06. Cambridge, MA, USA, pp. 153–160. .
[5] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (Mar. 2003) 993–1022.
[6] M.M. Bronstein, A.M. Bronstein, F. Michel, N. Paragios, Data fusion through cross-modality metric learning using similarity-sensitive hashing, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 3594–3601.
[7] D. Cao, X. He, L. Miao, Y. An, C. Yang, R. Hong, Attentive group recommendation, in: The 41st International ACM SIGIR Conference on Research Development in Information Retrieval. SIGIR '18, 2018.
[8] D. Cao, X. He, L. Miao, G. Xiao, H. Chen, J. Xu, Social-enhanced attentive group recommendation, IEEE Transactions on Knowledge and Data Engineering 33 (3) (2021) 1195–1209.
[9] W. Chong, D. Blei, F. Li, Simultaneous image classification and annotation, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1903–1910.
[10] Q. Cui, S. Wu, Q. Liu, W. Zhong, L. Wang, Mv-rnn: A multi-view recurrent neural network for sequential recommendation, IEEE Transactions on Knowledge and Data Engineering PP, 1–1. .
[11] M. del Carmen Rodríguez-Hernández, S. Ilarri, R.T. Lado, R. Hermoso, Location-aware recommendation systems: Where we are and where we recommend to go, in: Proceedings of the Workshop on Location-Aware Recommendations, LocalRec 2015, co-located with the 9th ACM Conference on Recommender Systems (RecSys 2015), Vienna, Austria, September 19, 2015. pp. 1–8. .
[12] A.M. Elkahky, Y. Song, X. He, A multi-view deep learning approach for cross domain user modeling in recommendation systems, in: Proceedings of the 24th International Conference on World Wide Web. WWW '15. pp. 278–288. .
[13] S. Feng, H. Zhang, J. Cao, Y. Yao, Merging user social network into the random walk model for better group recommendation, Appl. Intell. 49 (6) (2019) 2046–2058.
[14] Z. Feng, H. Li, W. Zeng, S. Yang, H. Qu, Topology density map for urban data visualization and analysis, IEEE Trans. Vis. Comput. Graph. 27 (2) (2021) 828–838.
[15] H. Gao, J. Tang, H. Liu, gscorr: modeling geo-social correlations for new check-ins on location-based social networks, in: 21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 – November 02, 2012. pp. 1582–1586. .

[16] L. Gao, H. Yang, J. Wu, C. Zhou, W. Lu, Y. Hu, Recommendation with multi-source heterogeneous information, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18. International Joint Conferences on Artificial Intelligence Organization, 7 2018., pp. 3378–3384.
[17] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, T.-S. Chua, Neural collaborative filtering, in: Proceedings of the 26th International Conference on World Wide Web. WWW '17, 2017.
[18] Z. He, C.-Y. Chow, J.-D. Zhang, Game: Learning graphical and attentive multi-view embeddings for occasional group recommendation, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 649–658, https://doi.org/10.1145/3397271.3401064.
[19] H. Hotelling, Relations between two sets of variates, Biometrika 28 (1936) 321–377.
[20] D. Kingma, J. Ba, Adam: A method for stochastic optimization, International Conference on Learning Representations. .
[21] D.P. Kingma, J. Ba, 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. .
[22] J.J. Levandoski, M. Sarwat, A. Eldawy, M.F. Mokbel, Lars: A location-aware recommender system, in: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering, ICDE '12. pp. 450–461. .
[23] H. Li, Y. Wang, Z. Lyu, J. Shi, Multi-task learning for recommendation over heterogeneous information network, IEEE Transactions on Knowledge and Data Engineering (2020) 1.
[24] Y. Li, M. Yang, Z. Zhang, Multi-view representation learning: A survey from shallow methods to deep methods, CoRR (2016), abs/1610.01206.
[25] X. Liu, Q. He, Y. Tian, W. Lee, J. McPherson, J. Han, Event-based social networks: linking the online and offline social worlds, in: The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12–16, 2012. pp. 1032–1040. .
[26] X. Liu, Y. Tian, M. Ye, W. Lee, Exploring personal impact for group recommendation, in: 21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 – November 02, 2012. pp. 674–683. .
[27] Z. Lu, H. Li, N. Mamoulis, D.W. Cheung, HBGG: a hierarchical bayesian geographical model for group recommendation, in: Proceedings of the 2017 SIAM International Conference on Data Mining, Houston, Texas, USA, April 27–29, 2017. pp. 372–380. .
[28] Z. Lu, H. Wang, N. Mamoulis, W. Tu, D.W. Cheung, Personalized location recommendation by aggregating multiple recommenders in diversity, GeoInformatica 21 (3) (Jul 2017) 459–484.
[29] R. Pan, Y. Zhou, B. Cao, N.N. Liu, R.M. Lukose, M. Scholz, Q. Yang, One-class collaborative filtering, in: Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15–19, 2008, Pisa, Italy. pp. 502–511. .
[30] S. Purushotham, C.J. Kuo, J. Shahabdeen, L. Nachman, Collaborative group-activity recommendation in location-based social networks, in: Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Crowdsourced and Volunteered Geographic Information, GeoCrowd '14, Dallas, Texas, USA, November 4, 2014. pp. 8–15. .
[31] Y. Qian, Z. Lu, N. Mamoulis, D. Cheung, P-lag: Location-aware group recommendation for passive users 07 (2017) 242–259.
[32] N. Srivastava, R. Salakhutdinov, Multimodal learning with deep boltzmann machines, J. Mach. Learn. Res. 15 (1) (Jan. 2014).
[33] S. Sun, A survey of multi-view machine learnig, Neural Comput. Appl. 23 (2014) 2031–2038.
[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L.U. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems 30. NIPS '17. pp. 5998–6008. .
[35] P. Vincent, H. Larochelle, Y. Bengio, P.-A. Manzagol, Extracting and composing robust features with denoising autoencoders, in: Proceedings of the 25th International Conference on Machine Learning. ICML '08. ACM, New York, NY, USA, pp. 1096–1103. http://doi.acm.org/10.1145/1390156.1390294 .
[36] C. Wang, D.M. Blei, Collaborative topic modeling for recommending scientific articles, in: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, August 21–24, 2011. pp. 448–456. .
[37] H. Wang, Z. Lu, Preference-aware sequence matching for location-based services, GeoInformatica 24 (1) (2020) 107–131.
[38] H. Wang, N. Wang, D.-Y. Yeung, Collaborative deep learning for recommender systems, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '15. New York, NY, USA, pp. 1235–1244 .
[39] R. Xu, J. Du, Z. Zhao, Y. He, Q. Gao, L. Gui, Inferring user profiles in social media by joint modeling of text and networks, Sci. China Inf. Sci. 62 (11), 219104:1–219104:3. .
[40] M. Ye, X. Liu, W. Lee, Exploring social influence for recommendation: a generative model approach, in: The 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '12, Portland, OR, USA, August 12–16, 2012. pp. 671–680. .
[41] H. Yin, Q. Wang, K. Zheng, Z. Li, J. Yang, X. Zhou, Social influence-based group representation learning for group recommendation, in: 2019 IEEE 35th International Conference on Data Engineering (ICDE), 2019, pp. 566–577.
[42] H. Yin, Q. Wang, K. Zheng, Z. Li, X. Zhou, Overcoming data sparsity in group recommendation, IEEE Transactions on Knowledge and Data Engineering (2020) 1.
[43] Q. Yuan, G. Cong, C. Lin, COM: a generative model for group recommendation, in: The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24–27, 2014. pp. 163–172. .
[44] W. Zeng, C.-W. Fu, S. Mller Arisona, S. Schubiger, R. Burkhard, K.-L. Ma, Visualizing the relationship between human mobility and points of interest, IEEE Transactions on Intelligent Transportation Systems 18 (8) (2017) 2271–2284.
[45] Y. Zhen, Y. Gao, D. Yeung, H. Zha, X. Li, Spectral multimodal hashing and its application to multimedia retrieval, IEEE Transactions on Cybernetics 46 (1) (2016) 27–38.