IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

SPECIAL SECTION ON INNOVATION AND APPLICATION OF INTELLIGENT PROCESSING OF DATA INFORMATION AND KNOWLEDGE AS RESOURCES IN EDGE COMPUTING

Received October 20, 2019, accepted November 7, 2019, date of publication November 12, 2019, date of current version November 22, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2953176*

# Group Recommendation via Self-Attention and Collaborative Metric Learning Model

**HAIYAN WANG[1,2,3], YULIANG LI[1,2], AND FELIX FRIMPONG[1]**
[1]School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
[2]Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing 210023, China
[3]Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing 210023, China

Corresponding author: Haiyan Wang (wanghy@njupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61772285, and in part by the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing.

**ABSTRACT** Group recommendation has attracted wide attention owing to its significance in real applications. One of the big challenges for group recommendation systems is how to integrate individual preferences of each group member and attain overall preferences for the group. Most of the traditional group recommendation solutions regard group members as equal participants and assign a same weight to each member. As a result, performance of this type of recommendation methods is not as good as expected. To improve the performance of group recommendation, a novel group recommendation model via Self-Attention and Collaborative Metric Learning (SACML) is presented in this paper. With the employment of Self-Attention mechanism, the SACML model can learn the similarity interactions between group members and services and decide a different weight for different group member. Based on these weights, group preferences for services can be generated by the aggregation of group members' preferences and the group's own preference. Similar metric space between group and services is obtained via collaborative metric learning with the group preferences and positive and negative services' features. Group recommendation is finally implemented based on the obtained metric space. Simulation has been conducted on CAMRa2011 and Meetup datasets, and experimental results show that the proposed SACML model has better performance in comparison with those baseline methods.

**INDEX TERMS** Self-attention, collaborative metric learning, group recommendation.

## I. INTRODUCTION

With the rapid development of information technology, the problem of information overload has become more and more serious. When faced with a large amount of information on the Internet, users find it hard to discover required and relevant information quickly and accurately according to their preferences. Recommendation systems have become one of the effective mitigation methods and an ideal recommendation method is supposed to be able to help users quickly find the services of interest. In real life, many daily activities are carried out in the form of groups [1]. For example, a group of friends planning to organize a dinner party, some scholars participating in an academic conference, and a group of customers who want to organize a group-buying activity. Traditional personalized recommendation methods fail to meet the demand of groups and as such, group-oriented recommendation is attracting more and more attention [2].

Different from personalized recommendation [3], group recommendation focuses on two key issues: 1) How to effectively aggregate group members' preferences; 2) How to make recommendations based on the aggregated group preferences. For the issues, our motivation is to improve performance of group recommendation by calculating the similarity between group members' preferences. We argue that the higher the similarity is, the bigger weight these members should be assigned. It will be easier to meet the needs of most group members during group recommendation based on the weighted integration of group preferences.

Most of the traditional group recommendation methods get group members' explicit preferences and regard different members as homogeneous objects in a static way. They assume group members have same weights during the aggregation of users' preferences. The preference aggregation strategies mainly include average strategy (AVG) [4],

The associate editor coordinating the review of this manuscript and approving it for publication was Ying Li.

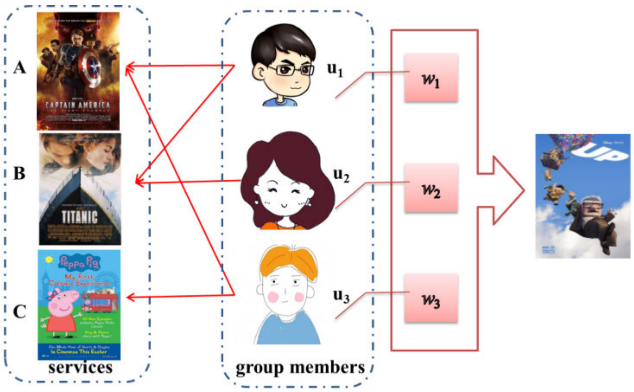



**FIGURE 1.** Scenario of a family movie recommendation.

least misery strategy (LM) [5], and maximum satisfaction strategy [6]. However, in a real application such as a family union or social activity, most of these group members are heterogeneous because of their age, gender, intelligence and personality. All which are factors that may affect a user's choice. When a user participates in a group decision-making process as a group member, main factors that need to be considered are the interactions between group members and services such as click-through rates, purchase records, and browsing records. For this reason, autonomous learning of the interactions, measuring the similarity between group members scientifically, and integrating the similarity weights with group members' preferences should be considered in the research field of group recommendation [7].

Nowadays, many applications provide service recommendation to groups. One of the difficulties faced by these recommendation systems is the mining interactive information between groups and services [8], [9]. As shown in Figure 1, in a family movie recommendation scenario, each family member has individual preference to each movie. Father u1 prefers movie A and B, mother u2 likes movie B and their son u3's favorite movies are A and C. Studying similarity between preferences of group members to assign reasonable weights for each group members and making group recommendations are the main work of this paper.

In this paper, we will discuss two key issues concerning how to obtain group recommendations.

The first issue focuses on building a model to learn the similarity of group members' preferences from the interaction information between members and services, with the purpose of integrating the preferences to obtain group preferences.

In recent years, neural network models such as recurrent neural network (RNN) and convolutional neural network (CNN) have become popular choices to solve the problems of sequence prediction and feature extraction [10]. In the RNN model, the interaction between service sequences with time characteristics is captured by a cyclic matrix, and the long-term dependencies are stored in memory. On the other hand, CNN captures implicit interactive information by the convolution kernel on the input sequence. When these two models are applied to group recommendation, neither of them can clearly capture the interaction between group members and services to get the similarity between group members. RNN cannot capture the integration in parallel. On the other hand, CNN can weakly capture the integration characteristics by the sliding-window concatenated transformations [11]. Based on these observations, this paper employs self-attention mechanism to learn the interaction information between group members and services, which can capture the long-dependent global interaction in parallel. In this way, we can adjust the aggregation strategy adaptively to generate group preferences.

Another problem here, is how to effectively use group preference vectors to improve the performance of group recommendation. For this purpose, we use collaborative metric learning to map group preference vectors and service feature vectors into the same embedding space for group recommendation. Specifically, CML calculates the similarity between group preference vectors and service feature vectors in the same embedding space, and ranks them in order to get the recommended service list. In this way, we improve the performance of group recommendation.

The main contributions of this paper are as follows:

1. This paper proposes the SACML model which combines the self-attention mechanism with metric learning, and implements group recommendation.

2. This paper develops a solution to compute the similarity weights of group members' preferences based on self-attention mechanism by using interaction between group members and services, which is aggregated with members' preferences to generate group preferences.

3. This paper conducts a series of experiments on public datasets to verify the validity of our proposed SACML model.

## II. RELATED WORK

In this section, we briefly review the related work from three aspects: group recommendation, self-attention mechanism and collaborative metric learning.

### A. GROUP RECOMMENDATION

Group recommendation is an important application problem in many social activities and industries such as online shopping, meal selection and travelling. Although personalized recommendation technologies such as collaborative filtering (CF) [12] and matrix factorization (MF) [13] have been widely studied, the research on group recommendation is still limited. The existing group recommendation methods are mainly focused on preference aggregation methods and score aggregation methods. Preference aggregation method uses the preferences of all group members to provide group recommendations [14], [15]. The score aggregation method combines the service recommendation score of each group member to get the group recommendation score. Among these two methods, two most popular aggregation strategies are AVG and LM. The AVG strategy regards the average score of group members on services as the final score of the group, but the recommendation generated by the average strategy may include services that are unfavorable to some members. The LM strategy pleases each member by choosing the lowest group member score as the final score, thus filtering out the low scores service which may cause disadvantages to some members. Since the LM strategy depends on the lowest scoring members, it may eventually recommend services that no member likes. Both of these methods are predefined strategies, which assign weight to group members statically. In real life, however, group members are mostly heterogeneous, and various factors will affect users' choice in the group. How to measure the similarity of group members' preferences as weights according to the interaction between group members and services and how to generate group preferences by integrating the weighted preferences are important issues in group recommendation.

This paper adopts the self-attention mechanism to measure the similarity between group members' preferences from their historical interactions, and assign dynamic weights to group members.

### B. SELF-ATTENTION MECHINSM

Attention mechanism in deep learning was first proposed in the field of visual image, similar to human visual attention mechanism. It pays attention to specific parts during training and has been widely used in many fields, such as natural language processing and computer vision [16], [17]. In recent years, some studies have also applied attention mechanism to recommendation systems [18]. For example, Da Cao et al. proposed a model based on the combination of attention mechanism and NCF model to solve the problem of group recommendation [5]. His paper is the first to propose a group recommendation algorithm using neural network model to learn aggregation strategy and also, to alleviate the problem of cold start. Tran D et al. also proposed a group recommendation model based on attention mechanism, which mainly aggregates the preferences of group members, and combines the aggregated group preferences with personalized ranking recommendation for group recommendation [6].

Self-attention mechanism was first proposed and used in machine translation by Google [19]. Unlike the traditional attention mechanism, self-attention mechanism focuses on interactive learning between multiple words in a sentence well [20]. Similarly, the self-attention mechanism can be used to learn the interaction between users and services in group recommendation. There are some methods to learn personalized interaction between users and services, such as matrix factorization [21] and Markov chain [22]. However, matrix factorization is hard to explain for implicit interaction and Markov chain needs too much computation. Self-attention mechanism takes into account the similarity relationship between group members' preferences, and obtains the interpretable weights of members rapidly [23]. This paper mainly uses self-attention mechanism to learn the similar interaction between members and services to aggregate the preferences of group members.

### C. COLLABORATIVE METRIC LEARNING

Metric learning algorithm can capture the important relationship between data by generating distance metrics. It is an important technology in many machine learning applications, like document retrieval, image classification, etc. The main idea is to allocate smaller distances between similar objects and larger distances between objects with fewer similarities [24]. For example, in service recommendation, we want to allocate small distances to services with similar user preferences, large distances to services with different preferences, and finally select the recommendation services according to the minimum distance.

Collaborative filtering (CF), as the mainstream idea of service recommendation system [9], mainly uses point product to measure the distance between user vector and service vector. The larger the point product, the closer the two vectors are [24]. However, the distance between point product and service vector does not satisfy the triangular inequality condition, that is, the distance can be transferred. For example, X is close to Y and X is close to Z, so Y is close to Z. Collaborative metric learning (CML) [25] is therefore preferred in this application.

Collaborative metric learning is a recently developed collaborative filtering algorithm. Its main idea is learning a user-service joint measure to encode the interactions between a user and services. Specifically, the measure will bring similar user-service pairs closer and push other dissimilar user-service pairs farther apart. Since CML satisfies the triangular inequality, this process will aggregate services similar to user preferences [26].

In this paper, we use CML to recommend services for groups, in which the group preference vectors obtained by self-attention mechanism and the expected service vectors are acquired in the training stage. As far as we know, this paper is the first group recommendation method that combines self-attention mechanism and collaborative metric learning.

## III. SACML MODEL

This paper proposes a group recommendation method based on Self-Attention Collaborative Metric Learning (SACML) to recommend services to all members of a fixed group. Generally speaking, the proposed model, SACML, comprises two parts: 1) group preference aggregation based on self-attention mechanism; 2) group recommendation based on collaborative metric learning. This paper first introduces the problem description of group recommendation in section A. Section B and C elaborate on two important parts of the model.

### A. PROBLEM DESCRIPTION

Let $G$, $U$ and $S$ be the sets of all groups, all users and all services, respectively.

$$G = \{g_1, g_2, \cdots, g_t, \cdots, g_{|G|}\}$$
$$U = \{u_1, u_2, \cdots, u_i, \cdots, u_{|U|}\}$$
$$S = \{s_1, s_2, \cdots, s_j, \cdots, s_{|S|}\}$$

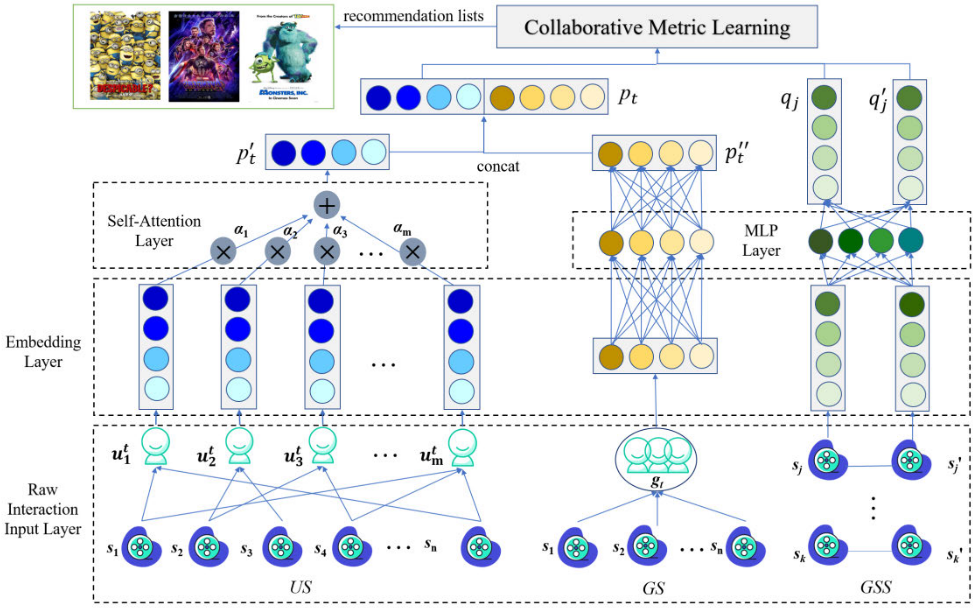



**FIGURE 2.** SACML model.

where $g_t$ denotes the $t^{\text{th}}$ group, $u_i$ denote the $i^{\text{th}}$ user and $s_j$ denote the $j^{\text{th}}$ service.

There are three observable interactions in the above three sets:

- Group-service interaction $GS = \{g_t\_s_j\}$, $1 \leq t \leq |G|$, $1 \leq j \leq |S|$, where $g_t\_s_j$ denotes the interaction between the $t^{\text{th}}$ group and the $j^{\text{th}}$ service, that is,

$$g_t\_s_j = \begin{cases} 1, & \text{if interaction } <g_t, s_j> \text{ exists} \\ 0, & \text{otherwise.} \end{cases}$$

- User-service interaction $US = \{u_i\_s_j\}$, $1 \leq i \leq |U|$, $1 \leq j \leq |S|$, where $u_i\_s_j$ denotes the interaction between the $i^{\text{th}}$ user and the $j^{\text{th}}$ service, that is,

$$u_i\_s_j = \begin{cases} 1, & \text{if interaction } <u_i, s_j> \text{ exists} \\ 0, & \text{otherwise.} \end{cases}$$

- Group-service positive and negative preference triples $GSS = \{(g_t, s_j, s_j')\}$, $1 \leq t \leq |G|$, $1 \leq j \leq |S|$, that is, service $s_j$ is a positive preference for group $g_t$, while service $s_j'$ is a negative one.
  Relevant symbol definitions are shown in Table 1.

With known $GS$, $US$ and $GSS$, the goal of this paper is to recommend a list of services to all members of the given group $g_t$, which contains $m$ members.

The SACML model is proposed for group recommendation in this paper.

As shown in figure 2, SACML mainly consists of four layers and a collaborative metric learning model. After three inputs $US$, $GS$ and $GSS$, the embedding layer is used to learn user preference, group self-preference and service feature respectively. With the help of user preference embedding, the similarity weights of group members in the group is calculated through Self-Attention mechanism. Group preference vector $p_t'$ will then be aggregated with group members' preferences and similarity weights. Group self-preference embedding and service embedding pass through the MLP layer, which will be used for feature space transformation and implicit relations extraction. After that, the group self-preference vectors $p_t''$, the positive service feature vectors $q_j$ and the negative one $q_j'$ will be obtained through MLP layer. The final group preference vector $p_t$ is obtained by concatenating the group preference aggregation vector $p_t'$ with the group self-preference vector $p_t''$. Through collaborative metric learning model, the group recommendation lists can be obtained.

### B. BASE MODEL

In this section, we mainly introduce two base deep learning layers, embedding layer and MLP layer.

#### 1) EMBEDDING

The embedding layer is designed to learn low-dimensional potential representation and reduce large-scale sparse interaction relationships' dimension at the same time [27]. In this paper, raw interaction relationships are $US \in \mathbb{R}^{|U|\times|S|}$, $GS \in \mathbb{R}^{|G|\times|S|}$ and $GSS \in \mathbb{R}^{|G|\times 2\times|S|}$. In order to transform large-scale sparse interaction relationships into low-dimensional embedding vectors, we multiply interaction relationships with three weight matrices $W_u \in \mathbb{R}^{|S|\times d}$, $W_g \in \mathbb{R}^{|S|\times d}$ and $W_s \in \mathbb{R}^{|S|\times d}$ respectively, where d is the embedding size. Three low-dimensional embedding vectors can be obtained, which are user embeddings $\boldsymbol{u}^t\{\boldsymbol{u}_1^t, \boldsymbol{u}_2^t, \cdots, \boldsymbol{u}_i^t, \cdots\} \in \mathbb{R}^{|U|\times d}$, group embeddings $\boldsymbol{g}\{\boldsymbol{g_1}, \boldsymbol{g_2}, \cdots, \boldsymbol{g_t}, \cdots\} \in \mathbb{R}^{|G|\times d}$ and paired service embeddings in groups $\boldsymbol{s}\{(s_j, s_j'), \cdots, (s_k, s_k')\} \in \mathbb{R}^{|G|\times 2\times d}$, where $\boldsymbol{u}_i^t$ means the $i^{th}$ user's embedding of the $t^{th}$ group, $\boldsymbol{g_t}$ means

**TABLE 1. Relevant symbol definitions.**

| Symbol | Definition | Symbol | Definition |
|---|---|---|---|
| $G$ | Set of all groups | $\mathbf{u}_i^t$ | Potential preference vector of the member $u_i$ in group $g_t$ |
| $U$ | Set of all users | $\mathbf{g}_t$ | The potential preference vector of the group $g_t$ |
| $S$ | Set of all services | $W_f$ | Weight parameter matrix in feature space $f$ |
| $g_t$ | the $t^{th}$ group | $W_h$ | Weight parameter matrix in feature space $h$ |
| $u_i$ | the $i^{th}$ user | $W_z$ | Weight parameter matrix in feature space $z$ |
| $s_j$ | the $j^{th}$ service | $d$ | the dimension of group preference vector and service feature vector. |
| $GS$ | Group-service interaction | $\beta_{ii'}$ | The self-attention model's focus on the attention extent to user $i'$ when moving to user $i$ |
| $US$ | User-service interaction | $\alpha_i$ | The weight of group member $u_i^t$ |
| $GSS$ | Group-service positive and negative preference triple set | $\gamma$ | A parameter regulating group aggregation preference and group self-preference |
| $g_{t\_}s_j$ | the interaction between the $t^{th}$ group and the $j^{th}$ service. | $p_t'$ | Group preference vector aggregated by group members |
| $u_{i\_}s_j$ | the interaction between the $i^{th}$ user and the $j^{th}$ service. | $p_t^*$ | Group self-preference |
| $(g_t, s_j, s_j')$ | the $t^{th}$ group preference triple with a positive service $s_j$ and a negative service | $p_t$ | Final group preference vector |
| $m$ | Number of group members | $q_j$ | The feature vector of the corresponding positive preference service $s_j$ in the group preference triple |
| $n$ | Number of services associated with groups | $q_j'$ | The feature vector of the corresponding negative preference service $s_j'$ in the group preference triple |
| $u_i^t$ | the $i^{th}$ member in group $g_t$ | $d(g_t, s_j)$ | Euclidean distances of $g_t$ and $s_j$ |

the $t^{th}$ group's embedding and $\boldsymbol{s}_j$ means the $j^{th}$ service's embedding.

### 2) MULTIPLE LAYER PERCEPTRON (MLP)

MLP (Multi-Layer Perceptron) is a forward-structured artificial neural network that maps a set of input vectors to a set of output vectors for feature space transformation and implicit relations extraction [27]. Group embedding and service embedding are fed into MLP with an activation function such as RELU. In this paper, we adopt two layers of full connection layer and the RELU function.

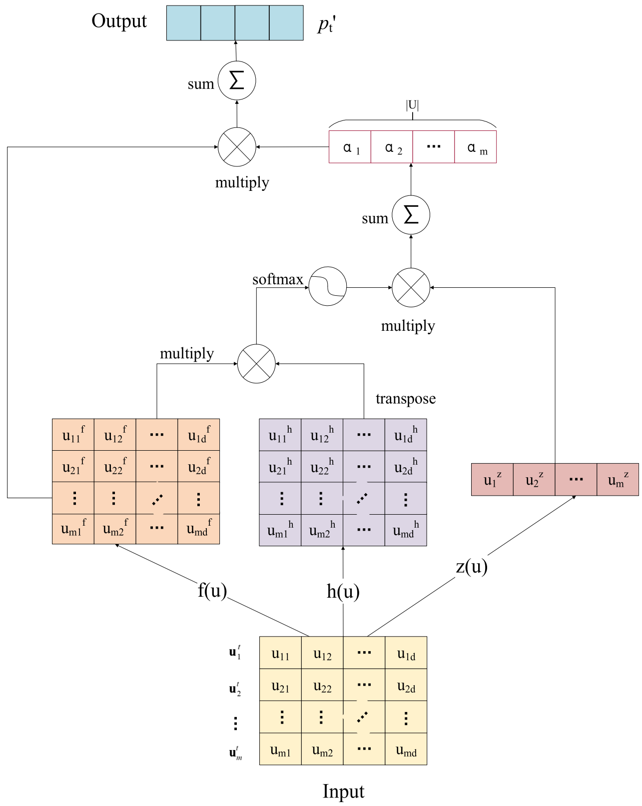



**FIGURE 3. Group preference aggregation process based on self-attention mechanism.**

## C. GROUP PREFERENCE AGGREGATION BY SELF-ATTENTION MECHANISM

Presently, the aggregation strategies of group members' preferences such as average and least misery neither shows the importance of users in the group nor take into account the interaction between each group member and a service, which has a dynamic impact on the weight of group members in any given group. This paper, therefore, adopts self-attention mechanism to capture similarity relationship between group members, which learns the weights of each member in a group to aggregate the preferences of group members.

The whole process of preference aggregation of group members based on self-attention mechanism is shown in Fig.3. To enable all users in the group to pay attention to each other's preferences, we designed three feature spaces that can implicitly learn the similarity of user preferences in a group.

The detailed aggregation process is as follows:

The potential preference embedding vector of group member $\mathbf{u}_i^t \in \mathbb{R}^{1\times d}$ is attained after the embedding layer, and then we transform the preference embedding vectors of group member into two feature spaces $f$ and $h$ to calculate the weight value, where $f(\mathbf{u}_i^t) = W_f \mathbf{u}_i^{tT}$ and $h(\mathbf{u}_i^t) = W_h \mathbf{u}_i^{tT}$.

$$\beta_{ii'} = \frac{\exp(x_{ii'})}{\sum_{i'=1}^{m} \exp(x_{ii'})} \tag{1}$$

$$x_{ii'} = f(\mathbf{u}_i^t) h(\mathbf{u}_{i'}^t)^T \tag{2}$$

where $\beta_{ii'}$ indicates the degree to which the model pays attention to the $i'^{th}$ group member when synthesizing the $i^{\text{th}}$ user space during the training process and $x_{ii'}$ represents the implicit similarity relationship between the $i'^{th}$ and $i^{\text{th}}$ group members' preferences in a group of two spaces learning together.

Let $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_m) \in \mathbb{R}^{m\times 1}$ the group member's weight, so the weight of the $i^{\text{th}}$ group member is

$$\alpha_i = \sum_{i'=1}^{m} \beta_{ii'} z(\mathbf{u}_i^t) \tag{3}$$

$$z(\mathbf{u}_i^t) = W_z \mathbf{u}_i^{tT} \tag{4}$$

In the formulas mentioned above, $W_f \in \mathbb{R}^{d\times d}$, $W_h \in \mathbb{R}^{d\times d}$, $W_z \in \mathbb{R}^{1\times d}$. These parameters are weight parameter matrices, where $d$ means dimension of feature space, $m$ is number of group members and $n$ shows the number of services in the group.

Therefore, the group members' preference vectors projected into the feature space are multiplied by the weight to aggregate the group preference vector as follows:

$$p_t' = \sum_{i=1}^{m} \alpha_i f(\mathbf{u}_i^t) \tag{5}$$

However, there are some scenarios which need to consider the group's own preferences for a better recommendation. For example, a work group with three members of A, B, and C are requested to see an advertising video, and A usually favors animated movies, B likes action movies, and C prefers comedy movies. Although they have their individual preferences for movies, they finally agree to see this movie because they belong to the same work group. From this example, we can conclude that it is very necessary to take group's self-preference into consideration for a more accurate group recommendation.

Let $p_t''$ be the self-preference vector of the group and $\mathbf{g}_t$ the potential preference embedding vector of the group. The self-preference of the group $p_t''$ is obtained through a full connection layer, as shown in formula (6):

$$p_t'' = \text{RELU}(V_g \mathbf{g}_t^T) \tag{6}$$

where $V_g \in \mathbb{R}^{d\times d}$ is the parameter matrix and $d$ is the feature dimension of a group preference embedding vector.

In order to combine the aggregated group preference with the group self-preference, a linear operation is performed.

$$p_t = \gamma p_t' + (1-\gamma) p_t'' \tag{7}$$

where $p_t$ represents the final group preference vector, $p_t'$ expresses the group aggregation preference vector and $p_t''$ means the group's self-preference vector. In order to adjust the proportion of $p_t'$ and $p_t''$, $\gamma$ is proposed. When $\gamma = 1$, the final group preference vector only considers the group aggregation preference vector $p_t'$.

### D. GROUP RECOMMENDATION BASED ON COLLABORATIVE METRIC LEARNING

In order to measure the similarity between group and service and realize group recommendations based on it, this paper trains a common metric space to encode the similarities between group preference vectors and service feature vectors by using Collaborative Metric Learning algorithm.

With the help of positive and negative group-service preference triple $(g_t, s_j, s_j')$, the hinge loss function is used to minimize the distance between similar group preference vector and service feature vector meanwhile maximizing the distance between dissimilar ones.

In this paper, we use $d(g_t, s_j)$ to represent the Euclidean distance between group $g_t$ and service $s_j$, which represents the group's preference for different services. Its value is equal to:

$$d(g_t, s_j) = \|p_t - q_j\| \tag{8}$$

where $p_t$ is the preference vector of the group $g_t$ and $q_j$ means the feature vector of the service $s_j$, which is a service embedding [28], acquired through embedded layer learning. It represents the inherent attributes of services.

The loss function obeys the relative preference of groups for different services, that is to say, the distance between groups and positive services is closer, and the distance between groups and negative services is farther. Therefore, we use a hinge loss function [25] to express such a constraint:

$$L_m = \sum_{v\in I_v} \sum_{v\notin I_v} \max(0, m + d(g_t, s_j)^2 - d(g_t, s_j')^2) \tag{9}$$

where $s_j$ is the positive service and $s_j'$ is the negative one. $g_t$ denotes the $t^{\text{th}}$ group, $m$ is the boundary, which means a tolerance of two distance difference.

According to the loss function $L_{\text{m}}$, each group's target neighbor is its favorite service, that is, positive service. For positive services, the gradient of loss function draws them closer to the group, while for negative services, the gradient pushes them away from the group until they come out of the boundary. Fig.4 illustrates the principle of group recommendation based on collaborative metric learning. There are four interactive services for group, including two positive services $s_1$, $s_2$ and two negative services $s_3$, $s_4$. The red arrow acts as the gradient. The left side of the graph is the state before model training, and the right side is the state after training. During CML model training, the positive services are pulled closer to the group by the gradient, and the negative services are pushed away until they exceed the margin of security limit.

Considering the interpretability of similarity value range, the similarity between the group preference vectors and service feature vectors is computed by the cosine similarity

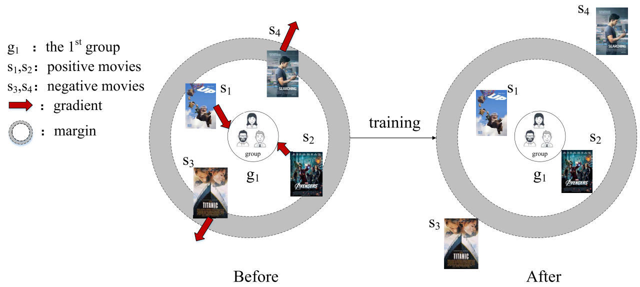



**FIGURE 4.** Group recommendation principle based on CML.

measure method [29], as shown in Formula (10):

$$c_{tj} = \frac{p_t \cdot q_j}{\|p_t\| \cdot \|q_j\|} \tag{10}$$

Finally, the cosine similarity value is sorted in ascending order, and the Top-K services are formed into recommendation lists [30].

In this paper, stochastic gradient descent is used to minimize hinge loss function of CML, and Adam optimization method to control learning rate. The parameter learning process of SACML model is as follows:

1) Input a positive and negative group-service preference triple $(g_t, s_j, s_j')$ and a boundary *margin*.

2) Update the group preference vector $p_t$ and the service feature vector $q_j$. Their gradients are calculated separately, and the learning rate is adjusted adaptively by Adam optimization method. This step is repeated until convergence.

3) Output and compute the cosine similarity between $p_t$ and $q_j$, and generate a recommendation list.

## IV. EXPERIMENTAL SIMULATION AND EFFECTIVENESS EVALUATION

In order to verify the validity and accuracy of SACML, this paper first introduces the experimental preparation of the whole experiment, and then shows the experimental results and analyses them.

### A. EXPERIMENTAL PREPARATION

This section will elaborate on the experimental settings, experimental data sets, evaluation criteria, baseline algorithm and parameter settings.

#### 1) EXPERIMENTAL SETTINGS

The experimental hardware environment is CPU Intel (R) Xeon (R) CPU E5-2630 v2@ 2.60 GHz, memory 128GB, and system Centos 7.0. In order to implement SACML algorithm, Python, a deep learning framework PyTorch and scientific computing libraries numpy and SciPy are used in this paper.

#### 2) EXPERIMENTAL DATA SETS

In order to test the effectiveness of the proposed group recommendation method for SACML, two popular public datasets CAMRa2011 [5] and Meetup datasets [6] are selected for experiments.

**TABLE 2.** Camra2011 data set statistics.

| Data object | Data attributes | Number of Data |
|---|---|---|
| #Groups | #number | 290 |
| | # avg group size | 2 |
| | #pos-neg | 1450 |
| #Users | #number | 601 |
| | #pos-neg | 3010 |
| #Services | #number | 7709 |

**TABLE 3.** Meetup data set statistics.

| Data object | Data attributes | Number of Data |
|---|---|---|
| #Groups | #number | 16330 |
| | #avg group size | 685 |
| | #group and service interactive information | 31214 |
| #Users | #number | 5893887 |
| | #user and service interactive information | 3195246 |
| #Services | #number | 2510 |

CAMRa2011 is a movie data set, which is grouped into training and test sets. Among them, the group data set contains the group members' relationships and also contains the positive and negative film relations of the family group; the user data of members includes the scores of a single family member on a single movie and the positive and negative film relative relations of each user member. The data statistics of the training data set are shown in Table 2.

The Meetup dataset comes from Meetup.com, an online social event service that helps people publish and participate in face-to-face events. The data set statistics used in this paper are shown in Table 3.

#### 3) EVALUATION CRITERIA

In this paper, four evaluation indicators are used to evaluate the performance of the model: precision [31], recall [31], AUC [32] and NDCG [33]. Let K be the number of recommendation services where K is set to 5,10 and 15.

For Top-K group recommendation services, the precision and recall formulas are defined as follows:

$$precision = \frac{|\{top - K\text{recommendations list}\} \cap \{\text{true items}\}|}{|\{top - K \text{ recommendations list}\}|} \tag{11}$$

$$recall = \frac{|\{top - K\text{recommendations list}\} \cap \{\text{true items}\}|}{|\{\text{true items}\}|} \tag{12}$$

AUC is an effective index to measure the quality of recommendation ranking. For each group, the following formula

**TABLE 4.** Performance comparison of difference group recommendation methods when K = 5.

| K=5 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Model | CAMRA2011 | | | | Meetup | | | |
| | Precision | Recall | AUC | NDCG | Precision | Recall | AUC | NDCG |
| CF_AVG | 0.066435 | 0.397109 | 0.203667 | 0.351926 | 0.140435 | 0.657109 | 0.343667 | 0.594726 |
| CF_LM | 0.058327 | 0.168054 | 0.155233 | 0.297622 | 0.075202 | 0.480254 | 0.312323 | 0.343622 |
| MF_AVG | 0.086714 | 0.307785 | 0.244296 | 0.334056 | 0.142898 | 0.664785 | 0.334296 | 0.587056 |
| MF_LM | 0.079216 | 0.253098 | 0.178196 | 0.210735 | 0.080216 | 0.503098 | 0.338996 | 0.361952 |
| CML_AVG | 0.103456 | 0.371457 | 0.293507 | 0.365671 | 0.125913 | 0.704529 | 0.356234 | 0.601653 |
| CML_LM | 0.094738 | 0.396831 | 0.304681 | 0.354973 | 0.136829 | 0.652896 | 0.390126 | 0.591468 |
| AGREE | 0.113574 | 0.583435 | 0.353862 | 0.395214 | 0.148382 | 0.763841 | 0.428654 | 0.648864 |
| SACML | 0.117485 | 0.586945 | 0.354641 | 0.400213 | 0.151974 | 0.776298 | 0.447326 | 0.651317 |

can be used:

$$AUC = \frac{S - M(M+1)/2}{M \times N} \tag{13}$$

where $M$ and $N$ denote the positive and negative number of movies respectively. In addition, $S = \sum R_j$, where $R_j$ represents the $j^{\text{th}}$ relevant services in the recommended list for group $g_t$.

The formulas for calculating NDCG evaluation criteria are as follows:

$$NDCG = \frac{DCG}{IDCG} \tag{14}$$

$$DCG = \sum_{i=1}^{K} \frac{rel_i}{\log_2(i+1)} \tag{15}$$

where in the formula of computing Discounted Cumulative Gain (DCG), $rel_i$ represents the degree of relevance of the $i^{\text{th}}$ recommendation service. When $rel_i = 1$, the $i^{\text{th}}$ recommendation service is selected by the group in a real environment, otherwise $rel_i = 0$. IDCG is a normalized DCG, that is, under a perfect sort, the maximum DCG value obtained.

#### 4) ALGORITHM BASELINE

- User-based CF with averaging strategy (CF-AVG) [6]: CF-AVG uses user-based collaborative filtering method to calculate each user's preference score for each candidate service, and then averages the preference score for all users to obtain the group recommendation score for each service.
- User-based CF with least-misery strategy (CF-LM) [6]: CF-LM is based on the least misery strategy. Similar to CF-AVG, CF-LM first uses user-based collaborative filtering to calculate each user's score for each candidate service. However, the lowest preference score of all users for the service is taken as the recommended score for each service.
- Matrix Factorization with averaging strategy (MF-AVG) [6]: MF-AVG is a matrix factorization model. MF-AVG takes the average score of all group members for a service as the recommended score of the service. Therefore, MF-AVG considers that all users in a group have equal weights of influence, that is, assume that all members contribute equally to the group.
- Matrix Factorization with least-misery strategy (MF-LM): Like MF-AVG, MF-LM is also an MF model, but the group's recommendation score for a service is derived from the lowest preference score of all users in the group.
- Collaborative metric learning with averaging strategy (CML_AVG) [25]: CML can learn a joint metric space to encode not only users' preferences but also the user-user and item-item similarity. Finally, group recommendations can be made by averaging user preferences and measuring similarities between the group and items.
- Collaborative metric learning with least-misery strategy (CML_LM) [25]: CML_LM is also a CML model with least-misery strategy.
- Attentive Group Recommendation (AGREE) [5]: The AGREE model adopts an attention mechanism to attain group preference, and learns the interaction between groups and items in the NCF framework.

#### 5) PARAMETER SETTINGS

For all baseline algorithms, this paper carries out experiments under the optimal parameter setting. For SACML model, the Gauss distribution stochastic initialization parameters with mean 0 and standard deviation 0.05 are used at first, and the loss functions are optimized by using Adam optimizer. The optimal batch values and learning rates are searched in [64, 128, 256, 512] and [0.0001, 0.001, 0.01, 0.1] respectively. Dropout rate is tuned amongst {0, 0.3, 0.5, 0.7}. The embedding dimensions of groups, users and services are set to 32, 32 and 64 respectively, and the ReLU activation function is added after the embedding layer.

### B. EXPERIMENTAL RESULTS

#### 1) RESULTS OF EVALUATION INDICATORS

This section compares the predicted results of SACML model with those of baseline models on two datasets. Table 4,

**TABLE 5.** Performance comparison of difference group recommendation methods when K = 10.

| K=10 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Model | CAMRa2011 | | | | Meetup | | | |
| | Precision | Recall | AUC | NDCG | Precision | Recall | AUC | NDCG |
| CF_AVG | 0.041649 | 0.469158 | 0.327109 | 0.427953 | 0.074946 | 0.753965 | 0.352219 | 0.619546 |
| CF_LM | 0.039351 | 0.265177 | 0.298054 | 0.395328 | 0.050137 | 0.501467 | 0.401964 | 0.351836 |
| MF_AVG | 0.057959 | 0.472638 | 0.323085 | 0.377069 | 0.070619 | 0.708347 | 0.419368 | 0.605723 |
| MF_LM | 0.046524 | 0.329472 | 0.317798 | 0.261487 | 0.058326 | 0.623462 | 0.474267 | 0.501267 |
| CML_AVG | 0.059158 | 0.572583 | 0.341073 | 0.411876 | 0.076549 | 0.792833 | 0.507109 | 0.609553 |
| CML_LM | 0.061842 | 0.442318 | 0.347162 | 0.409674 | 0.065613 | 0.789351 | 0.490054 | 0.572128 |
| AGREE | 0.069321 | 0.781445 | 0.427231 | 0.457162 | 0.082959 | 0.803734 | 0.507785 | 0.630269 |
| SACML | 0.071836 | 0.791366 | 0.429557 | 0.460182 | 0.088216 | 0.810513 | 0.529023 | 0.657516 |

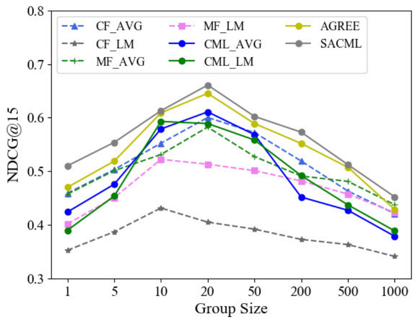
**TABLE 6.** Performance comparison of difference group recommendation methods when K = 15.

| K=15 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Model | CAMRa2011 | | | | Meetup | | | |
| | Precision | Recall | AUC | NDCG | Precision | Recall | AUC | NDCG |
| CF_AVG | 0.029448 | 0.481246 | 0.336784 | 0.438235 | 0.034678 | 0.768347 | 0.395844 | 0.620382 |
| CF_LM | 0.027951 | 0.372367 | 0.307655 | 0.402967 | 0.026802 | 0.564182 | 0.406839 | 0.405266 |
| MF_AVG | 0.023487 | 0.533696 | 0.338646 | 0.388654 | 0.030266 | 0.713875 | 0.452167 | 0.582356 |
| MF_LM | 0.028065 | 0.428749 | 0.326478 | 0.303656 | 0.032185 | 0.649156 | 0.432509 | 0.513261 |
| CML_AVG | 0.032856 | 0.626726 | 0.395435 | 0.432355 | 0.040448 | 0.796726 | 0.508458 | 0.610725 |
| CML_LM | 0.030146 | 0.603249 | 0.380458 | 0.419065 | 0.039479 | 0.790342 | 0.475435 | 0.589065 |
| AGREE | 0.037849 | 0.851746 | 0.442182 | 0.485321 | 0.043827 | 0.815346 | 0.592182 | 0.645321 |
| SACML | 0.038624 | 0.852536 | 0.452602 | 0.487493 | 0.048672 | 0.822553 | 0.604526 | 0.660678 |

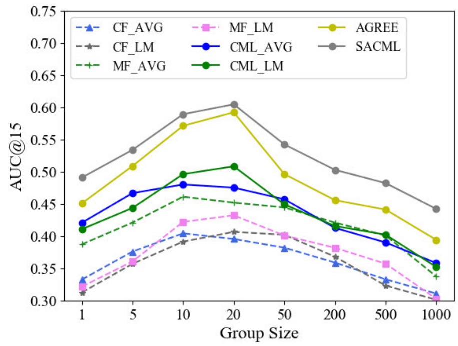
Table 5 and Table 6 compare Precision, Recall, AUC and NDCG values of various methods under K = 5, 10 and 15 respectively.

It can be seen from the tables that:

- Indicators of SACML model perform best amongst the baseline models compared. This proves the validity of the SACML model in this paper. More specifically, self-attention mechanism plays an active role in aggregating group members' preferences and providing recommendation services for groups. The SACML model uses deep learning to render a better performing recommender system in comparison to the traditional recommendation methods (CF, MF and CML). This shows the superiority of the neural network, especially in simulating the high-dimensional spatial interaction among users, groups and services. Compared with the deep learning model AGREE, the SACML recommendation algorithm also has better performance. SACML learns the similar relationships among group members, which is able to better glean the interests of the group as a whole. AGREE model's attention layer just focuses on preference relationships between group members and services rather than similarities between group members, and the inputs of the training and prediction phases are different.
- Among all the methods, Precision decreases with the increase of the number of recommended services K, while Recall, AUC and NDCG increase. Because Precision and Recall are two contradictory measures, generally speaking, when Precision is high, Recall tends to be low, otherwise the opposite is true. The reason why Precision is relatively low is that there is only one group-preferred service in CAMRa2011 and Meetup data. The more recommended services there are, the lower the Precision.
- From the overall effect point of view, the six score aggregation baseline algorithms on the four evaluation indicators do not always maintain optimal effect. For example, in the case of K = 5, the NDCG evaluation index of CML_AVG is better than that of CML_LM, but CML_AVG's AUC evaluation index is worse than that of CML_LM's. This indicates that the recommendation effect of the traditional score aggregation baseline method is not stable, possibly because the weight given to group members' preferences is fixed, while the




(a)Group Size -NDCG




(b)Group Size -AUC

**FIGURE 5.** Model performances for different group sizes on meetup dataset.

weight given to each group member by SACML is set to different values according to the interaction relationship of members in the group, which shows the flexibility and superiority of SACML model.

- Compared with four evaluation indicators in two datasets CAMRa2011 and Meetup, due to the large amount of data, there are more interaction behaviors between users and services in groups, and the recommendation performances of all models in Meetup is better than those in CAMRa2011.

### 2) THE INFLUENCE OF EXPERIMENTAL PARAMETERS

#### *a: PERFORMANCE OF DIFFERENT GROUP SIZE*

In order to study the performance of each recommendation method under different group sizes, the Meetup dataset is used to test the performance of eight groups with the sizes of 1, 5, 10, 20, 50, 200, 500, 1000 respectively. Fig.5 shows the calculated NDCG@15 and AUC@15 curves.

Fig.5 shows that SACML achieves better performance than other baseline methods at different group sizes. When the group size is less than 10, the performance of the model becomes better and better with the increase of the group size, because group members provide more interactive information and the results of recommendation are more accurate. When the group size is between 10 and 20, the group recommendation performance reaches the best. When the group size increases again, the recommendation performance decreases due to the high diversity in the group.

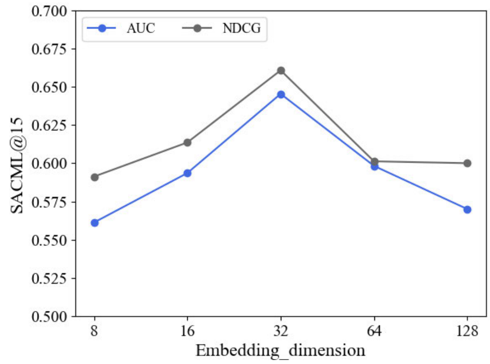



**FIGURE 6.** SACML performances for different embedding dimension on meetup dataset.

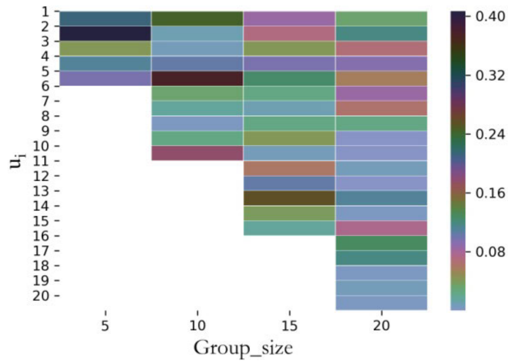



**FIGURE 7.** Attention weights of members in group with different group size learned by SACML.

#### *b: PERFORMANCE OF EMBEDDING DIMENSION*

Fig.6 shows the performance of the SACML model using Meetup data set in different dimensions of embedding vectors for groups, users and services. The NDCG@15 and AUC@15 curves are shown below. In the graph, larger dimensions do not necessarily lead to higher model performance, because high-dimensional space can easily lead to over-fitting. Smaller dimension cannot attain better performance due to its less representation.

#### *c: WEIGHT VISUALIZATION*

The SACML model allows us to calculate the weights of each member in a group for further analysis. In this paper, four groups of sizes 5, 10, 15 and 20 were randomly selected to learn weights. Fig.7 shows that the X axis represents the four groups' size, the Y axis represents the member $u_i$ corresponding to each group, and the color depth represents the weight value. The analysis of weight visualization may make the recommendation results of SACML model interpretable. In Fig.7, if the weight of the member $u_2$ is the largest (darkest color) in the group size of 5, i.e., the member may be the leader in the group, and the group recommendation results can be explained by the preference of that leader.

## V. CONCLUSION

In this paper, we propose a new recommendation model SACML, which combines neural network and metric learning. It uses self-attention mechanism to automatically learn the importance of each group member from the interaction between group members and services, while the group preferences are aggregated. We then use collaborative metric learning to implement group recommendations. In order to verify the validity of SACML, two real data sets were tested. The results show that compared with the seven baseline algorithms, SACML achieves better performance in group recommendation.

In the follow-up work, we will carry out in-depth research in the following two directions: 1) The self-attention network can dig out potential groups according to the interaction between users, and then recommend to these groups. Using self-attention network to carry out group discovery research, therefore, has good practical value. 2) SACML is developing to incorporate contextual information such as social relations, text information (like commodity reviews) and time information into the model. Also seeing that group member preferences are time variant, it will be of great research value to consider how to use time context information to realize online group recommendations.

## REFERENCES

[1] J. Xu, Z. Yan, G. Cao, and J. Zhao, ‘‘Family shopping recommendation system using behavior sequence data and user profile,’’ in *Proc. 10th Int. Conf. Internet Multimedia Comput. Service (ICIMCS)*, New York, NY, USA, 2018, pp. 41:1-41:7.

[2] H. Yin, Q. Wang, K. Zheng, Z. Li, J. Yang, and X. Zhou, ‘‘Social influence-based group representation learning for group recommendation,’’ in *Proc. IEEE 35th Int. Conf. Data Eng. (ICDE)*, Apr. 2019, pp. 566–577.

[3] S. Yun, R. Kim, M. Ko, and J. Kang, ‘‘SAIN: Self-Attentive Integration Network for Recommendation,’’ in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2019, pp. 1205–1208, doi: 10.1145/3331184.3331342.

[4] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, ‘‘Neural graph collaborative filtering,’’ in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2019, pp. 165–174, doi: 10.1145/3331184.3331267

[5] J. Shi, B. Wu, and X. Lin, ‘‘A latent group model for group recommendation,’’ in *Proc. IEEE Int. Conf. Mobile Services*, New York, NY, USA, 2015, pp. 233–238, doi: 10.1109/MobServ.2015.41.

[6] D. Cao, X. He, L. Miao, Y. An, C. Yang, and R. Hong, ‘‘Attentive group recommendation,’’ in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2018, pp. 645–654, doi: 10.1145/3209978.3209998.

[7] L. V. Tran, T.-A. N. Pham, Y. Tay, Y. Liu, G. Cong, and X. Li, ‘‘Interact and decide: Medley of sub-attention networks for effective group recommendation,’’ in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2019, pp. 255–264, doi: 10.1145/3331184.3331251.

[8] D. Sacharidis, ‘‘Group recommendations by learning rating behavior,’’ in *Proc. 25th Conf. User Modeling, Adapt. Pers. (UMAP)*, 2017, pp. 174–182, doi: 10.1145/3079628.3079691.

[9] J. Zhao, K. Liu, and F. Tang, ‘‘A group recommendation strategy based on user's interaction behavior,’’ in *Proc. IEEE 2nd Inf. Technol., Netw., Electron. Autom. Control Conf. (ITNEC)*, Chengdu, China, Dec. 2017, pp. 1170–1174, doi: 10.1109/ITNEC.2017.8284960.

[10] S. Zhang, L. Yao, A. Sun, and Y. Tay, ‘‘Deep learning based recommender system: A survey and new perspectives,’’ *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, 2019, doi: 10.1145/3285029.

[11] R. He and J. McAuley, ‘‘Fusing similarity models with Markov chains for sparse sequential recommendation,’’ in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Dec. 2016, pp. 191–200.

[12] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, ‘‘Attentive collaborative filtering: Multimedia recommendation with item- and component-level attention,’’ in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2017, pp. 335–344, doi: 10.1145/3077136.3080797.

[13] L. Boratto, S. Carta, A. Chessa, M. Agelli, and M. L. Clemente, ‘‘Group recommendation with automatic identification of users communities,’’ in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol.*, Milan, Italy, 2019, pp. 547–550, doi: 10.1109/WI-IAT.2009.346.

[14] Z. Yujie and D. Yulu, ‘‘Research on group recommendation system and their application,’’ *Chin. J. Comput.*, vol. 39, no. 4, pp. 745–764, 2016.

[15] X. Chen, H. Xu, Y. Zhang, J. Tang, Y. Cao, Z. Qin, and H. Zha, ‘‘Sequential recommendation with user memory networks,’’ in *Proc. 11th ACM Int. Conf. Web Search Data Mining (WSDM)*, Feb. 2018, pp. 108–116.

[16] Q. Zhu, X. Zhou, Z. Song, J. Tan, and L. Guo, ‘‘DAN: Deep attention neural network for news recommendation,’’ in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 5973–5980, doi: 10.1609/aaai.v33i01.33015973.

[17] C. Zhou, J. Bai, J. Song, X. Liu, Z. Zhao, X. Chen, and J. Gao, ‘‘ATRank: An attention-based user behavior modeling framework for recommendation,’’ in *Proc. Assoc. Adv. Artif. Intell. (AAAI)*, 2018, pp. 4564–4571.

[18] X. Tang, T. Wang, and H. Yang, ‘‘AKUPM: Attention-enhanced knowledge-aware user preference model for recommendation,’’ in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 1891–1899, doi: 10.1145/3292500.3330705.

[19] V. Ashish, S. Noam, P. Niki, U. Jakob, J. Llion, N. A. Gomez, K. Lukasz, and P. Illia, ‘‘Attention is all you need,’’ in *Proc. Annu. Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 5998–6008.

[20] S. Zhang, Y. Tay, L. Yao, and A. Sun, ‘‘Next item recommendation with self-attention,’’ 2018, *arXiv:1808.06414*. [Online]. Available: https://arxiv.org/abs/1808.06414

[21] S. Zhang, L. Yao, Y. Tay, X. Xu, X. Zhang, and L. Zhu, ‘‘Metric factorization: Recommendation beyond matrix factorization,’’ 2018, *arXiv:1802.04606*. [Online]. Available: https://arxiv.org/abs/1802.04606

[22] C. Cai, R. He, and J. McAuley, ‘‘SPMC: Socially-aware personalized Markov chains for sparse sequential recommendation,’’ in *Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2017, pp. 1476–1482, doi: 10.24963/ijcai.2017/204.

[23] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, ‘‘Self-attention generative adversarial networks,’’ 2018, *arXiv:1805.08318*. [Online]. Available: https://arxiv.org/abs/1805.08318

[24] H. Gao, W. Huang, Y. Duan, X. Yang, and Q. Zou, ‘‘Research on cost-driven services composition in an uncertain environment,’’ *J. Internet Technol.*, vol. 20, no. 3, pp. 755–769, 2019.

[25] C.-K. Hsieh, L. Yang, Y. Cui, T.-Y. Lin, S. Belongie, and D. Estrin, ‘‘Collaborative metric learning,’’ in *Proc. 26th Int. Conf. World Wide Web (WWW)*, 2017, pp. 193–201, doi: 10.1145/3038912.3052639.

[26] Y. Tay, L. A. Tuan, and S. C. Hui, ‘‘Latent relational metric learning via memory-based attention for collaborative ranking,’’ in *Proc. World Wide Web Conf. (WWW)*, 2018, pp. 729–739, doi: 10.1145/3178876.3186154

[27] Y. Feng, F. Lv, W. Shen, M. Wang, F. Sun, Y. Zhu, and K. Yang, ‘‘Deep session interest network for click-through rate prediction,’’ in *Proc. 28th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2019, pp. 2301–2307.

[28] A. Tamhane, S. Arora, and D. Warrier, ‘‘Modeling contextual changes in user behaviour in fashion e-commerce,’’ in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*, in Lecture Notes in Computer Science, vol. 10235. Cham, Switzerland: Springer, Apr. 2017, pp. 539–550, doi: 10.1007/978-3-319-57529-2_42.

[29] A. Heidarian and M. J. Dinneen, ‘‘A hybrid geometric approach for measuring similarity level among documents and document clustering,’’ in *Proc. IEEE 2nd Int. Conf. Big Data Comput. Service Appl. (BigDataService)*, Oxford, U.K., 2016, pp. 142–151, doi: 10.1109/BigDataService.2016.14.

[30] H. Gao, H. Miao, L. Liu, J. Kai, and K. Zhao, ‘‘Automated quantitative verification for service-based system design: A visualization transform tool perspective,’’ *Int. J. Softw. Eng. Knowl. Eng.*, vol. 28, no. 10, pp. 1369–1397, 2018, doi: 10.1142/S0218194018500390.

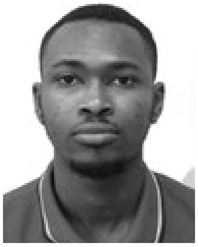
[31] F. Lv, T. Jin, C. Yu, F. Sun, Q. Lin, K. Yang, and W. Ng, ‘‘SDM: Sequential deep matching model for online large-scale recommender system,’’ in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, Sep. 2019, doi: 10.1145/3357384.3357818

[32] J. Gorla, N. Lathia, S. Robertson, and J. Wang, ‘‘Probabilistic group recommendation via information matching,’’ in *Proc. 22nd Int. World Wide Web Conf. (WWW)*, Rio de Janeiro, Brazil, May 2013, pp. 495–504

[33] C. Xu, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, F. Zhuang, J. Fang, and X. Zhou, ‘‘Graph contextualized self-attention network for session-based recommendation,’’ in *Proc. 28th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2019, pp. 3940–3946.

**HAIYAN WANG** received the Ph.D. degree from the School of Computer Science, Nanjing University of Posts and Telecommunications, China. She is currently a Professor with the School of Computer Science, Nanjing University of Posts and Telecommunications. She is also the Deputy Director of Jiangsu Key Laboratory of Big Data Security and Intelligent Processing and Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks. Her research interests mainly focus on service computing, trusted computing, and big data intelligent processing technology. She has hosted two National Natural Science Foundation of China projects and ever received the second prize twice for progress in Jiangsu Science and Technology.

**YULIANG LI** is currently a master's degree with the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, School of Computer Science, Nanjing University of Posts and Telecommunications, China. Her main research interests include service computing, deep learning, and group recommendation.



**FELIX FRIMPONG** was born in 1996. He received the bachelor's degree from the Nanjing University of Posts and Telecommunications. He is currently pursuing the master's degree with the School of Computer Science and Technology, Donghua University. His main research interests include data mining, deep learning, and group recommendation.