Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence

Deep Modeling of Group Preferences for Group-Based Recommendation

Liang Hu[1,2], Jian Cao[1,*], Guandong Xu[2], Longbing Cao[2], Zhiping Gu[3], Wei Cao[2]

[1]Shanghai Jiaotong University, [2]University of Technology Sydney, [3]Shanghai Technical Institute of Electronics & Information
{lianghu,cao-jian}@sjtu.edu.cn, {guandong.xu,longbing.cao}@uts.edu.au, guzhiping@stiei.edu.cn, wei.cao@student.uts.edu.au

Abstract

Nowadays, most recommender systems (RSs) mainly aim to suggest appropriate items for individuals. Due to the social nature of human beings, group activities have become an integral part of our daily life, thus motivating the study on group RS (GRS). However, most existing methods used by GRS make recommendations through aggregating individual ratings or individual predictive results rather than considering the collective features that govern user choices made within a group. As a result, such methods are heavily sensitive to data, hence they often fail to learn group preferences when the data are slightly inconsistent with predefined aggregation assumptions. To this end, we devise a novel GRS approach which accommodates both individual choices and group decisions in a joint model. More specifically, we propose a deep-architecture model built with collective deep belief networks and dual-wing restricted Boltzmann machines. With such a deep model, we can use high-level features, which are induced from lower-level features, to represent group preference so as to relieve the vulnerability of data. Finally, the experiments conducted on a real-world dataset prove the superiority of our deep model over other state-of-the-art methods.

Introduction

In recent years, various recommender systems (RSs) have been applied to capture personalized requirements and offer tailored services for better user experience and new business opportunities. Collaborative filtering (CF) algorithms as a fundamental building block of modern RS have been widely studied. However, human beings are of a social nature, so various kinds of group activities are observed throughout life, e.g. seeing a movie, or planning travel. Recently, the RS community has begun to study group behavior to make group recommendations (Jameson and Smyth 2007). For instance, PolyLens (O'Connor et al. 2002) is an early GRS where users can create groups and ask for recommendations. Moreover, Berkovsky and Freyne (2010) studied recipe recommendations for families where all members eat a meal together.

Each member of a group may have different opinions on the same items, so the main challenge in GRSs is to satisfy most group members with diverse preferences. Obviously, this is not achieved through an individual-based CF method. To date, the existing mainstream approaches of GRS try to aggregate group information from individual user models (Jameson and Smyth 2007, Masthoff 2011). In general, these methods can be classified into two types of models which differ in the timing of data aggregation. The first type of model is called *Group Preference Aggregation* (GPA), which firstly aggregates all members' ratings into a group profile, and then any individual-based CF approach can be used if it regards groups as virtual individual users. In contrast, the second type of model is called *Individual Preference Aggregation* (IPA) which firstly predicts the individual ratings over candidate items, and then aggregates the predicted ratings of members within a group via predefined strategies to represent group ratings.

However, both these two aggregation models have their deficiencies. Quite often, only a few members will give ratings to the same items used by a group. Hence, we can hardly construct a representative group profile because each group rating is often generated merely from a single member. As a result, the recommendations may be biased towards some of the members in a group. The IPA model aggregates group recommendations using the predictive results produced by individual models. However, the recommendations made by individual models fail to consider individual behavior when she/he makes choices in the role of a group member. In essence, these models lack the capability to build a good representation of the group preference, which we believe crucial to the success of GRSs.

The above discussions disclose the need for a GRS to satisfy each member and the group as a whole. In this paper, we attempt to address this challenge by employing the deep-learning technique that has been proved effective to learn high-level features (Bengio et al. 2013). Since the hypothesis behind our approach is that the individual choices are governed by collective factors when she/he acts as a group member, we design a collective Deep Belief Network (DBN (Hinton et al. 2006)) to disentangle collective features w.r.t. a group, according to the choices of each member. Furthermore, each group choice can be regarded as a joint decision by all members, so we can take advantage of collective features as the priors to model the probability of making each group choice. Accordingly, we design a dual-wing Restricted Boltzmann Machine (RBM (Hinton et al. 2006)) at the top level to learn the representation of group preferences by jointly modeling group choices and collective features.

In summary, our main contributions include:

- We propose a deep-architecture model to learn a high-level representation of group preferences, which avoids the vulnerability of data in traditional approaches.
- We design a collective DBN over all member profiles of a group so as to disentangle the high-level collective features from the low-level members' features.
- We devise a dual-wing RBM at the top level to learn a comprehensive representation of group preferences using both collective features and group choices.
- We conducted empirical evaluations on a real-world data set. The results demonstrate the superiority of our approach in comparison with state-of-the-art methods.

Related Work

Since most current GRSs still employ individual-based CF techniques, we firstly review some state-of-the-art CF methods. The k-nearest neighborhood algorithm is an early CF method (Su and Khoshgoftaar 2009) which has been applied in some real-world RSs (Sarwar et al. 2001). However, this method does not work well when the data is very sparse because it may fail to find similar users or items. Recently, latent factor models have become the most prevalent approach in CF. Therein, matrix factorization (MF) methods have become dominant in recent years (Salakhutdinov and Mnih 2008, Koren et al. 2009). Recent developments have demonstrated the power of RBMs, which is able to extract useful features from input data. Hence, some researchers have studied CF with RBMs (Salakhutdinov et al. 2007, Georgiev and Nakov 2013). However, individual-based CF approaches cannot be directly used by GRSs because they assume that choices are independently made by individuals. In contrast, group-based choices are joint decisions made by all group members.

Current GRSs measure group satisfaction by means of aggregating members' information using some aggregation models, such as GPA and IPA. In fact, quite a few heuristic strategies have been designed to work with the aggregation models. In particular, *Average* and *Least Misery* are the two most prevalent strategies (Masthoff 2011), so they will be employed in this paper. For example, PolyLens (O'Connor et al. 2002) uses the *Least Misery* strategy which assumes a group tends to be as happy as its least happy member. Yu et al. (2006) took the *Average* strategy to recommend television programs for groups. Moreover, Berkovsky and Freyne (2010) compared these two strategies for recipe recommendations for families. In this paper, we study a case of movie recommendations for households, which was sponsored by CAMRa2011 (Said et al. 2011). Some recent work (Hu et al. 2011, Gorla et al. 2013) studied this problem using some aggregation models. Therein, Hu et al. (2011) tested the MF method under GPA and IPA models with various strategies. However, such methods are heavily dependent on the input data, which often fail to learn the representation of group preference when the data is slightly inconsistent with the aggradation assumption. To avoid such vulnerability, we design a deep model to represent group preference using high-level features that are learned from lower-level features. Such a deep model can effectively remove the sensitivity from data (Bengio et al. 2013).

Preliminaries

Firstly, we formulate the problem and introduce some concepts used in this paper. Then, we give a brief review on the RBM model and parameter estimation since our model is built with RBMs and DBNs. In fact, RBMs are the building blocks of a DBN, where the key idea is to use greedy layer-wise training (Hinton et al. 2006).

Problem Statement

This paper is aimed to learn an expressive representation of the group preferences so as to make appropriate recommendations to groups. Especially, we address the typical case of movie recommendation for households which was sponsored by CAMRa2011 Challenge (Said et al. 2011).

Before introducing our model, we first need to give some definitions to clarify the following presentation.

- ***Collective Features***: these represent compromised preferences of a group, which are shared among all members and can be disentangled from the *Member Features*.
- ***Individual Features***: these represent independent individual-specific preference, which can be disentangled from the *Member Features* w.r.t. this user.
- ***Member Features***: these model the individual preference of a user when she/he makes choices as a group member, which can be regarded as a mixture of *Collective Features* and *Individual Features*.

Restricted Boltzmann Machines

An RBM (Hinton et al. 2006) is a Markov random field over a vector of binary visible units $\boldsymbol{v} \in \{0,1\}^D$ and hidden units $\boldsymbol{h} \in \{0,1\}^F$, where the connections only exist between $\boldsymbol{v}$ and $\boldsymbol{h}$. The distribution of an RBM is defined through an energy function $\mathrm{E}(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta})$:

$$P(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) = \exp\big(-\mathrm{E}(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta})\big) / Z(\boldsymbol{\theta}) \qquad (1)$$

$\boldsymbol{\theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{d}\}$ are the model parameters, where $\boldsymbol{W} \in \mathbb{R}^{D\times F}$ encodes the visible-hidden interaction, $\boldsymbol{b} \in \mathbb{R}^D$ and $\boldsymbol{d} \in \mathbb{R}^F$ encodes the biases of $\boldsymbol{v}$ and $\boldsymbol{h}$. The pattern of such interaction can be formally specified through the energy function:

$$\mathrm{E}(\boldsymbol{v}, \boldsymbol{h}; \boldsymbol{\theta}) = -\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{v} - \boldsymbol{d}^{\mathrm{T}}\boldsymbol{h} \qquad (2)$$

The conditional distributions w.r.t. visible units and hidden units are factorial (Bengio et al. 2013), which can be easily derived from Eq. (1):

$$P(v_i = 1|\boldsymbol{h};\boldsymbol{\theta}) = s\left(b_i + \sum_{j=1}^{D} W_{ij}h_j\right) \quad (3)$$

$$P(h_j = 1|\boldsymbol{v};\boldsymbol{\theta}) = s\left(d_j + \sum_{i=1}^{K} v_i W_{ij}\right) \quad (4)$$

where $s(\cdot)$ is a sigmoid function. In particular, the RBM has been generalized to Gaussian RBM (GRBM) to work with real-value data. The energy of the GRBM is defined by:

$$\mathrm{E}(\boldsymbol{v},\boldsymbol{h};\boldsymbol{\theta}) = \sum_{i=1}^{D}\frac{(v_i-b_i)^2}{2\sigma_i^2} - \sum_{i=1}^{F} d_j h_j - \sum_{i=1}^{D}\sum_{j=1}^{F}\frac{v_i W_{ij} h_j}{\sigma_i} \quad (5)$$

where the Gaussian visible units $\boldsymbol{v} \in \mathbb{R}^D$, the hidden units $\boldsymbol{h} \in \{0,1\}^F$ and $\boldsymbol{\theta} = \{\boldsymbol{W},\boldsymbol{b},\boldsymbol{d},\boldsymbol{\sigma}\}$ are the model parameters. Accordingly, the conditional distributions w.r.t. each visible unit and each binary hidden unit are given by:

$$P(v_i|\boldsymbol{h};\boldsymbol{\theta}) = \mathcal{N}\left(b_i + \sigma_i \sum_{j=1}^{F} W_{ij}h_j, \sigma_i^2\right) \quad (6)$$

$$P(h_j = 1|\boldsymbol{v};\boldsymbol{\theta}) = s\left(d_j + \sum_{i=1}^{D} v_i W_{ij}/\sigma_i\right) \quad (7)$$

Each model parameters $\theta_k \in \boldsymbol{\theta}$ can be estimated using gradient descent to minimize the negative log-likelihood:

$$-\frac{\partial \log p(\boldsymbol{v};\boldsymbol{\theta})}{\partial \theta_k} = \mathbb{E}_{\boldsymbol{P}(\boldsymbol{h}|\boldsymbol{v})}\left(\frac{\partial \mathrm{E}(\boldsymbol{v},\boldsymbol{h};\boldsymbol{\theta})}{\partial \theta_k}\right) - \mathbb{E}_{\boldsymbol{P}(\boldsymbol{v},\boldsymbol{h})}\left(\frac{\partial \mathrm{E}(\boldsymbol{v},\boldsymbol{h};\boldsymbol{\theta})}{\partial \theta_k}\right) \quad (8)$$

The first term, a.k.a. data-dependent expectation, is tractable but the second term, a.k.a. model-dependent expectation, is intractable and must be approximated (Bengio et al. 2013). In practice, contrastive divergence (CD) (Hinton 2002) is a successful algorithm which approximates the expectation with a short k-step Gibbs chain (often k=1), denoted as CD_k. Moreover, Tieleman (2008) proposed an improved CD algorithm, namely persistent CD.

Model and Inference

Most current GRSs are built on GPA or IPA models, so they are vulnerable to the data. To address this issue, we need to learn high-level and abstract features to replace the shallow features that directly couple on data.

To learn high-level features, we build a multi-layer model in terms of a deep learning technique. Using such a model, we can recover low-level features accounting for the data, and then pool low-level features to form higher-level invariant features (Bengio et al. 2013). In particular, we employ DBN and RBM as the building blocks to construct a collective DBN, where the term "collective" signifies that this DBN jointly model all members in a group as a whole. This collective DBN is capable of disentangling the collective features from low-level member features. Such collective features are an abstract representation of group preference, which avoids the deficiency of direct aggregation on the individual ratings. Furthermore, we design a dual-wing RBM on the top of the DBN to learn the comprehensive features w.r.t. each group, where one wing is connected to the group profile and the other is connected to the collective features learned from the collective DBN. Such a deep-structure design jointly models group choices and collective features, so that it can produce high-level features to represent the group preference so as to overcome the vulnerabilities in current shallow models.

Disentangling Collective and Individual Features

In individual-based RSs, users independently make decisions on choosing which items, whereas in GRSs, each member needs to consider other members' preferences when he/she makes choices. That is, each choice is a mixed individual and collective decision. Therefore, we need to disentangle the individual and collective factors leading to the decisions.

To achieve this goal, we can first learn the low-level member features from the member profile, i.e. ratings given by the member, through the bottom-layer model depicted on the left of Figure 1. Then, we can disentangle the high-level collective and individual features from the member features using the top-layer model. In particular, the top-layer model is a collective RBM as illustrated on the right of Figure 1, where the plate notation is used to represent the repeated individual and member features of a group, and the collective features are coupled with all member features.

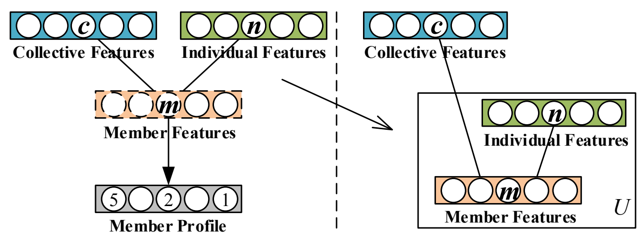

Figure 1: Left: Overview of the two-layer collective DBN used to disentangle high-level collective and individual features. Right: More detailed structure of the collective RBM at the top layer where the collective features are connected to the member features w.r.t. each member.

To date, the most effective approach to learn the parameters of a DBN is through greedy layer-wise training using a stack of RBMs (Hinton et al. 2006, Bengio 2009). In our model, the bottom-layer model is a GRBM w.r.t. each user. Such a user-based RBM model has been studied in the literature for individual-based CFs (Salakhutdinov et al. 2007, Georgiev and Nakov 2013). We simply use the same method to learn the member features, denoted $\boldsymbol{m}_u \in \mathbb{R}^D$, w.r.t. each member u, where the conditional distributions used for CD have been given by Eq. (6) and (7).

When the member features are learned, we take them as the visible units to learn higher-level features. In particular, it is possible to disentangle the collective and individual features from the member features since they represent a compromised preference among all members. As shown in Figure 1, we construct a collective RBM for each group, where the collective features, denoted $\boldsymbol{c} \in \mathbb{R}^K$, are connected to the member features w.r.t. each member. Then, we can write the following energy function to describe the interaction pattern of this collective RBM.

$$\mathrm{E}(\boldsymbol{m},\boldsymbol{n},\boldsymbol{c};\boldsymbol{\theta}) = -\boldsymbol{f}^T\boldsymbol{c} - \sum_{u=1}^{U}(-\boldsymbol{m}_u^T\boldsymbol{W}\boldsymbol{n}_u - \boldsymbol{m}_u^T\boldsymbol{X}\boldsymbol{c} - \boldsymbol{b}^T\boldsymbol{m}_u - \boldsymbol{d}^T\boldsymbol{n}_u)$$

where U denotes the number of members in the group and $\boldsymbol{\theta} = \{\boldsymbol{W},\boldsymbol{X},\boldsymbol{b},\boldsymbol{d},\boldsymbol{f}\}$ are the model parameters. $\boldsymbol{W} \in \mathbb{R}^{D\times F}$ encodes the interaction between member features and individual features and $\boldsymbol{X} \in \mathbb{R}^{D\times K}$ encodes the interaction between member features and collective features.

Similar to the conditional distributions for a standard RBM, we can easily derive the conditional distribution w.r.t. each member feature $m_{u,i}$, each individual feature $n_{u,j}$ and each collective feature c_k.

$$P(m_{u,i} = 1|\boldsymbol{c},\{\boldsymbol{n}_u\};\boldsymbol{\theta}) = s\left(b_i + \sum_{j=1}^{F} W_{ij}n_j + \sum_{k=1}^{K} X_{ik}c_k\right) \quad (9)$$

$$P(n_{u,j} = 1|\boldsymbol{m};\boldsymbol{\theta}) = s\left(d_j + \sum_{i=1}^{D} m_{u,i}W_{ij}\right) \quad (10)$$

$$P(c_k = 1|\boldsymbol{m};\boldsymbol{\theta}) = s\left(f_k + \sum_{u=1}^{U}\sum_{i=1}^{D} m_{u,i}X_{ij}\right) \quad (11)$$

With these conditional distributions in hand, we can learn each parameter $\theta_i \in \boldsymbol{\theta}$ using CD. For example, the stochastic gradient descent update using CD$_k$ is given by:

$$\theta_i \leftarrow \theta_i - \alpha\left(\frac{\partial\mathrm{E}(\boldsymbol{m}^0,\boldsymbol{n}^0,\boldsymbol{c}^0;\boldsymbol{\theta})}{\partial\theta_i} - \frac{\partial\mathrm{E}(\boldsymbol{m}^k,\boldsymbol{n}^k,\boldsymbol{c}^k;\boldsymbol{\theta})}{\partial\theta_i}\right) \quad (12)$$

where $\boldsymbol{m}^0$ are the visible data, $\boldsymbol{n}^0$ and $\boldsymbol{c}^0$ are respectively sampled from Eq. (10) and (11). $\boldsymbol{m}^k,\boldsymbol{n}^k,\boldsymbol{c}^k$ are the k-step sample from a Gibbs chain with the initial values $\boldsymbol{m}^0,\boldsymbol{n}^0,\boldsymbol{c}^0$. When the model parameters are learned, we set the value of collective feature c_k using its expectation, i.e. $\hat{c}_k = s\left(f_k + \sum_{m=1}^{M}\sum_{i=1}^{D} m_{m,i}X_{ij}\right)$, instead of a stochastic binary value to avoid unnecessary sampling noise (Hinton 2012).

Modeling a Comprehensive Representation of Group Preferences

GPA models create group profiles by aggregating individual ratings but, as discussed in the introduction, the recommendations may be biased towards a minority of members' taste based on such group profiles. To avoid such deficiency, the group profiles used in our approach simply consist of the group choices over items. Formally, we denote the group choices using binary ratings: $r_{gi} = 1$ indicates item i which was chosen by group g and $r_{gi} = 0$ otherwise. Given such group profiles, we can run an individual-based CF method for making recommendations by taking each group as a virtual user. However, only using such group profiles may lead to learning less expressive features because we cannot distinguish the degree of like on the same items between groups due to the identical ratings.

Each group choice is a joint decision made by all members whereas collective features exactly represent compromised preference of a group according to members' choices. Hence we can take advantage of the collective features to model the degree of like on an item, more formally, the probability of making that choice. As a result, we design a dual-wing RBM (DW-RBM) on the top of our model as illustrated in Figure 2, where one wing of the DW-RBM is connected to the group profile, and the other wing is connected to the collective features layer of the collective DBN. Under such a construction, it learns a set of comprehensive features that jointly model the group choices and the collective features. In fact, our approach can be viewed as a transfer learning model in which the collective features learned from the low-level collective DBN are transferred to the high-level DW-RBM model so as to learn a more comprehensive representation of the group preferences.

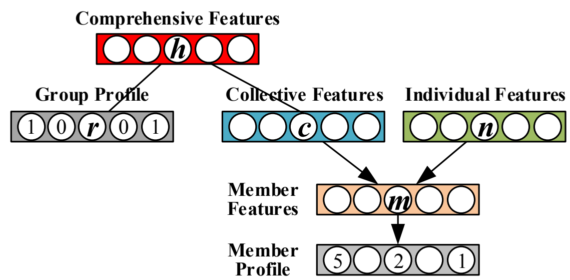

Figure 2: A dual-wing RBM is placed on the top of DBN, which jointly models the group choices and collective features to learn the comprehensive features of group preference.

For any item with a one-rating, i.e. $r_{gi} = 1$, we can say that group g is explicitly interested in item i. However, it is not certain that group g is not interested in item i or is unaware of it if $r_{gi} = 0$. Therefore, we cannot simply treat unchosen items as true-negative instances. Thus, it is a so-called "one-class" or "implicit feedback" CF problem (Hu et al. 2008, Pan et al. 2008). These methods use a weighted matrix factorization approach where it assigns a relatively large weight to apply a higher penalty on the loss on fitting one-ratings and a much smaller weight to apply a lower penalty on the loss on fitting zero-ratings (Hu et al. 2008, Pan et al. 2008). Equally, it can be interpreted from a probabilistic view (Wang and Blei 2011): the one-rating is generated from an informative distribution with a high confidence level whereas the zero-rating is generated from a less informative distribution with a low confidence level. Following the same idea (Wang and Blei 2011), we define a concentrated distribution governed by a small variance parameter for one-ratings whereas a diffuse distribution governed by a large variance parameter for zero-ratings.

$$\begin{cases}\sigma_{gi}^2 = \alpha f(g,i) & if\ r_{gi} = 1\\ \sigma_{gi}^2 = \beta & if\ r_{gi} = 0\end{cases} \quad (13)$$

where $\beta > \alpha f(i) > 0$, α, β are constants and the function $f(g,i)$ can simply be a constant 1, or a more sophisticated form to retrieve the group satisfaction measured by some aggregation strategy. For example, if we take the *Least Misery* strategy, we can define $f(g,i) = 1/lm(g,i)$, where

$lm(g, i)$ returns the least member rating on item i. That is, larger group satisfaction means smaller variance.

Following the setting of one-class CF (Wang and Blei 2011), we model the group profile using Gaussian visible units with different variance parameters. Under such a construction, the energy function for the DW-RBM can be defined as follows (note that we omit the subscript g for concise, since each DW-RBM models a single group):

$$\mathrm{E}(\boldsymbol{r}, \boldsymbol{c}, \boldsymbol{h}; \boldsymbol{\theta}) = \sum_{i=1}^{D} \frac{(r_i - b_i)^2}{2\sigma_i^2} - \sum_{k=1}^{K} f_k c_k - \sum_{j=1}^{Y} d_j h_j - \sum_{i=1}^{M} \sum_{j=1}^{Y} \frac{r_i W_{ij} h_j}{\sigma_i} - \sum_{k=1}^{K} \sum_{j=1}^{Y} c_k X_{kj} h_j \quad (14)$$

where $\boldsymbol{r} \in \{0,1\}^M$ are the group ratings, $\boldsymbol{h} \in \{0,1\}^Y$ are the comprehensive features and $\boldsymbol{\theta} = \{\boldsymbol{W}, \boldsymbol{X}, \boldsymbol{d}, \boldsymbol{f}\}$ are the model parameters. $\boldsymbol{W} \in \mathbb{R}^{M \times Y}$ encodes the interaction between $\boldsymbol{r}$ and $\boldsymbol{h}$ and $\boldsymbol{X} \in \mathbb{R}^{D \times F}$ encodes the interaction between $\boldsymbol{c}$ and $\boldsymbol{h}$. According to this energy function, we can respectively obtain the conditional distribution w.r.t. each rating r_i, each collective feature c_k, and each comprehensive feature h_j:

$$P(r_i | \boldsymbol{h}; \boldsymbol{\theta}) = \mathcal{N}\left(b_i + \sigma_i \sum_{j=1}^{F} W_{ij} h_j, \sigma_i^2\right) \quad (15)$$

$$P(c_k = 1 | \boldsymbol{y}; \boldsymbol{\theta}) = s\left(f_k + \sum_{j=1}^{Y} X_{kj} h_j\right) \quad (16)$$

$$P\left(h_j = 1 \middle| \boldsymbol{v}, \boldsymbol{c}; \boldsymbol{\theta}\right) = s\left(d_j + \sum_{i=1}^{M} r_i W_{ij} / \sigma_i + \sum_{k=1}^{K} c_k X_{kj}\right) \quad (17)$$

Then, the model parameters $\boldsymbol{\theta}$ can be estimated by CD as demonstrated in the previous subsection.

Recommendation for a Group

The one-class CF approach (Hu et al. 2008, Pan et al. 2008) ranks the items for recommendation according to the reconstructed ratings. Given a zero-rating item, the reconstructed rating tends to be relatively large if this item meets a user's preference, otherwise it tends to be small.

In the same way, we can reconstruct a group profile using the DW-RBM. In particular, we perform a one-step mean-field reconstruction (Welling and Hinton 2002) instead of a stochastic reconstruction to avoid sampling noise.

$$\hat{h}_j = s\left(d_j + \sum_{i=1}^{M} r_i W_{ij} / \sigma_i + \sum_{k=1}^{K} c_k X_{kj}\right) \quad (18)$$

$$\hat{r}_i = \mathbb{E}\left[\mathcal{N}\left(b_i + \sigma_i \sum_{j=1}^{F} W_{ij} \hat{h}_j, \sigma_i^2\right)\right] = b_i + \sigma_i \sum_{j=1}^{F} W_{ij} \hat{h}_j \quad (19)$$

Then, we can rank the recommendation items $\boldsymbol{C}$ for a group by sorting their reconstructed ratings $\{\hat{r}_i\}_{i \in \boldsymbol{C}}$.

Experiments

Many studies on GRS were evaluated using synthetic group preferences created from individual profiles due to the lack of available data on group preferences. However, such synthetic datasets cannot truly reflect the characteristics of group behaviors because all the individual choices are made independently. To overcome this deficiency, CAMRa2011 (Said et al. 2011) released a real-world dataset containing the movie watching records of households and the ratings on each watched movie given by some group members. Following track 1 of CAMRa2011, we evaluated our approach and other comparative methods to compare the performance of movie recommendation for households.

Data Preparation

The dataset for track 1 of CAMRa2011 has 290 households with a total of 602 users who gave ratings (on a scale 1~100) over 7,740 movies. This dataset has been partitioned into a training set and an evaluation set. The training set contains 145,069 ratings given by those 602 members, and 114,783 movie choice records from the view of 290 groups. That is, only 1.26 members give rating to a watched movie. The evaluation set contains 286 groups with 2,139 group-based choices. Some statistical information is provided in Table 1.

Table 1: Statistics of the evaluation data

Data	#Users/#Groups	#Ratings	Density
$\text{Train}_{\text{user}}$	602	145,069	0.0313
$\text{Train}_{\text{group}}$	290	114,783	0.0510
$\text{Eval}_{\text{group}}$	286	2,139	/

Evaluation Metrics and Comparative Methods

We use the metrics Mean Average Precision (MAP) and Area Under the ROC Curve (AUC) to evaluate models.

- ***MAP*** computes the mean of the average precision scores over all households $\boldsymbol{H}$

  $$MAP = \frac{1}{|\boldsymbol{H}|} \sum_{h \in \boldsymbol{H}} \frac{1}{|\boldsymbol{M}_h|} \sum_{m=1}^{|\boldsymbol{M}_h|} \frac{m}{\hat{r}(\boldsymbol{M}_{h,m})}$$

  where $\boldsymbol{M}_h$ denotes the relevant movies w.r.t. household h and $\hat{r}\left(\boldsymbol{M}_{h,m}\right)$ denotes the rank of the *m*-th relevant movie.
- ***AUC*** measures the probability that the rank of relevant movies $\boldsymbol{M}^+$ is higher than irrelevant movies $\boldsymbol{M}^-$ w.r.t. a group, and it is estimated as follows:

  $$AUC = \frac{\sum_{i \in \boldsymbol{M}^+} \sum_{k \in \boldsymbol{M}^-} \delta[rank(i) < rank(k)]}{|\boldsymbol{M}^+| \cdot |\boldsymbol{M}^-|}$$

  where δ(.) returns 1 if *rank(i)<rank(k)* and 0 otherwise.

To compare our approach with state-of-the-art methods, we evaluate the following methods in the experiments:

- *kNN*: This is a baseline method to recommend movies watched by the top-k most similar groups.
- *MF-GPA*: This method performs matrix factorization (Salakhutdinov and Mnih 2008) on the group ratings that are aggregated from individual ratings through a specified strategy.
- *MF-IPA*: This method performs matrix factorization on individual ratings, and then aggregates the predicted ratings as the group ratings, using a specified strategy.
- *OCMF*: This method performs one-class MF (Hu et al. 2008) on the binary group ratings where the weights are set according to a specified strategy.

Table 2: MAP and mean AUC of all comparative models with different strategies

	MAP			AUC		
Model/Strategy	*No Strategy*	*Average*	*Least Misery*	*No Strategy*	*Average*	*Least Misery*
kNN (k=5)	0.1595	N/A	N/A	0.9367	N/A	N/A
MF-GPA	N/A	0.1341	0.0628	N/A	0.9535	0.9297
MF-IPA	N/A	0.1952	0.1617	N/A	0.9635	0.9503
OCMF	0.2811	0.2858	0.2801	0.9811	0.9813	0.9810
OCRBM	0.2823	0.2922	0.2951	0.9761	0.9778	0.9782
DLGR	**0.3236**	**0.3252**	**0.3258**	**0.9880**	**0.9892**	**0.9897**

- *DLGR*: This is our deep learning approach, where the variance parameters of the DW-RBM (cf. the previous section) are set according to a specified strategy.
- *OCRBM*: This simply uses an RBM over the group choices without a connection to collective features. The variance parameters are set the same as the DW-RBM.

In the experiments, we tune the hyper parameters for each model, e.g. the dimensionality of latent features and the regularization parameters, by cross validation.

Results

To perform a comprehensive comparison, we evaluated all comparative methods using the two most prevalent, *Average* and *Least Misery,* aggregation strategies (if applicable), in addition to the evaluation without using any strategy. Specially, we set $\beta = 1$ and $\alpha = 0.5$ (cf. Eq. (13)) for OCRBM and DLGR when no strategy is used, and we set $\alpha = 1$ and $f(g, i) = 1/[1 + \log s(g, i)]$ when a strategy $s(\cdot)$ is used. Also, we used similar settings for the weights of OCMF.

The results of MAP and mean AUC are reported in Table 2. The baseline method kNN does not achieve a good performance because it is hard to find a set of groups with identical taste over a sparse dataset. For the similar reason, MF-IPA and MF-GPA also do not perform very well. Note that MF-IPA outperforms MF-GPA. The main reason for this is that most movies are rated by only one instead of most members, so the GPA model aggregates a biased group profile. OCMF and OCRBM perform much better than MF-IPA and MF-GPA because they construct their models on the group choices instead of the aggregated ratings but use them in a more subtle way. Moreover, it is easy to see that our model DLGR marginally outperforms any other method regardless of using an aggregation strategy or not using a strategy. The main reason is that all these methods except DLGR try to directly learn a good representation of the group preference from the data. However, they may fail to learn the expressive features based on such a shallow structure. In contrast, DLGR can learn a high-level representation from low-level features through deep architecture which removes the vulnerabilities of data. In particular, DLGR outperforms its sub-model OCRBM. This is because OCRBM makes no use of the individual member choices which contain useful features determining the group choices. In comparison, DLGR provides a more robust solution which not only models the group choices but also takes advantage of the collective features learned from all members' choices.

In general, a group with more members implies more different preferences, so it is harder to find recommendations satisfying all members. In our problem, each household may contain 2~4 members in this dataset. A household with 2 members, typically a couple, may easily agree on choosing a movie, whereas a household with more than 2 members, typically parents and children, may have different tastes due to the generation gap. Therefore, we additionally evaluated the MAP w.r.t. 2-member households and the 2^+-member households under *Average* and *Least Misery* strategies.

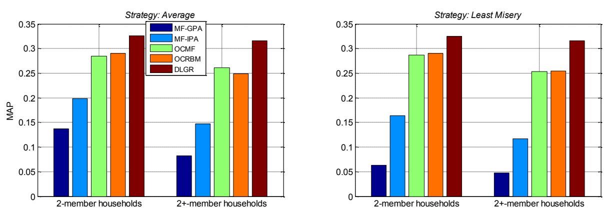

Figure 3: MAP w.r.t. 2-member groups vs. 2^+-member groups

Figure 3 plots the MAP w.r.t. the above two cases. We can see that DLGR outperforms all other comparative methods and the performance difference between the two cases is relatively small. Such a result proves that DLGR is still effective to represent group preference even when there are more members with different preferences. In comparison, other comparative models are constructed in a shallow manner, and are more sensitive to data hence they cannot learn the best features to represent group preference when the group becomes larger.

Conclusion

In this paper, we propose a deep learning approach to overcome the deficiencies in current GRSs. Essentially, our model aims to learn high-level comprehensive features to represent group preference so as to avoid the vulnerabilities in a shallow representation. The empirical evaluation on a real-world dataset proves that our approach can achieve much better performance than other state-of-the-art models. Since our approach constructs a deep architecture that is able to disentangle group-specific features at a high level, it is applicable to many other areas that study the group behavior with coupled interactions among members.

Acknowledgements

This work is partially supported by China National Science Foundation (Granted Number 61272438), Research Funds of Science and Technology Commission of Shanghai Municipality (12511502704). Australia-China Science & Research Fund (ACSRF15722).

*Jian Cao is the corresponding author

References

Bengio, Y.: Learning Deep Architectures for AI. *Found. Trends Mach. Learn. 2*, 1-127 (2009)

Bengio, Y., Courville, A., Vincent, P.: Representation Learning: A Review and New Perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 35*, 1798-1828 (2013)

Berkovsky, S., Freyne, J.: Group-based recipe recommendations: analysis of data aggregation strategies. In *Proceedings of the fourth ACM conference on Recommender systems*, pp. 111-118. ACM (2010)

Georgiev, K., Nakov, P.: A non-IID Framework for Collaborative Filtering with Restricted Boltzmann Machines. In *Proceedings of the 30th International Conference on Machine Learning*. JMLR: W&CP (2013)

Gorla, J., Lathia, N., Robertson, S., Wang, J.: Probabilistic group recommendation via information matching. In *Proceedings of the 22nd international conference on World Wide Web*, pp. 495-504. International World Wide Web Conferences Steering Committee (2013)

Hinton, G.: A Practical Guide to Training Restricted Boltzmann Machines. In Montavon, G., Orr, G., Müller, K.-R. (eds.) *Neural Networks: Tricks of the Trade*, vol. 7700, pp. 599-619. Springer Berlin Heidelberg (2012)

Hinton, G.E.: Training Products of Experts by Minimizing Contrastive Divergence. *Neural Computation 14*, 1771-1800 (2002)

Hinton, G.E., Osindero, S., Teh, Y.-W.: A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation 18*, 1527-1554 (2006)

Hu, X., Meng, X., Wang, L.: SVD-based group recommendation approaches: an experimental study of Moviepilot. In *Proceedings of the 2nd Challenge on Context-Aware Movie Recommendation*, pp. 23-28. ACM (2011)

Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. pp. 263-272. IEEE (2008)

Jameson, A., Smyth, B.: Recommendation to Groups. In Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) *The Adaptive Web*, vol. 4321, pp. 596-627. Springer Berlin Heidelberg (2007)

Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. *Computer 42*, 30-37 (2009)

Masthoff, J.: Group Recommender Systems: Combining Individual Models. In Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) *Recommender Systems Handbook*, pp. 677-702. Springer US (2011)

O'Connor, M., Cosley, D., Konstan, J.A., Riedl, J.: PolyLens: a recommender system for groups of users. In *ECSCW 2001*, pp. 199-218. Springer (2002)

Pan, R., Zhou, Y., Cao, B., Liu, N.N., Lukose, R., Scholz, M., Yang, Q.: One-class collaborative filtering. In *Eighth IEEE International Conference on Data Mining*, pp. 502-511. IEEE (2008)

Said, A., Berkovsky, S., Luca, E.W.D.: Group recommendation in context. In *Proceedings of the 2nd Challenge on Context-Aware Movie Recommendation*, pp. 2-4. ACM (2011)

Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In *Advances in neural information processing systems*, pp. 1257-1264. (2008)

Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted Boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*, pp. 791-798. ACM (2007)

Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pp. 285-295. ACM (2001)

Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. *Adv. in Artif. Intell. 2009*, 2-2 (2009)

Tieleman, T.: Training restricted Boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th international conference on Machine learning*, pp. 1064-1071. ACM (2008)

Wang, C., Blei, D.M.: Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 448-456. ACM (2011)

Welling, M., Hinton, G.: A New Learning Algorithm for Mean Field Boltzmann Machines. In Dorronsoro, J. (ed.) *Artificial Neural Networks — ICANN 2002*, vol. 2415, pp. 351-357. Springer Berlin Heidelberg (2002)

Yu, Z., Zhou, X., Hao, Y., Gu, J.: TV Program Recommendation for Multiple Viewers Based on user Profile Merging. *User Model User-Adap Inter 16*, 63-82 (2006)