Incorporating User Preferences across Multiple Topics into Collaborative Filtering for Personalized Merchant Recommendation

Yunfeng Chen[1], Lei Zhang[2], Xin Li[3], Yu Zong[4], Guiquan Liu[1](✉), and Enhong Chen[1]

[1] University of Science and Technology of China, Hefei, China
cyunfeng@mail.ustc.edu.cn, {gqliu,cheneh}@ustc.edu.cn
[2]Anhui University, Hefei, China
zl@ahu.edu.cn
[3] iFlyTek Research, Hefei, China
xinli2@iflytek.com
[4]West Anhui University; Texas A&M University, United States
Nick.zongy@gmail.com

Abstract. Merchant recommendation, namely recommending personalized merchants to a specific customer, has become increasingly important during the past few years especially with the prevalence of Location Based Social Networks (LBSNs). Although many existing methods attempt to address this task, most of them focus on applying the conventional recommendation algorithm (e.g. Collaborative Filtering) for merchant recommendation while ignoring harnessing the hidden information buried in the users' reviews. In fact, the information of user real preferences on various topics hidden in the reviews is very useful for personalized merchant recommendation. To this end, in this paper, we propose a graphical model by incorporating user real preferences on various topics from user reviews into collaborative filtering technique for personalized merchant recommendation. Then, we develop an optimization algorithm based on a Gaussian model to train our merchant recommendation approach. Finally, we conduct extensive experiments on two real-world datasets to demonstrate the efficiency and effectiveness of our model. The experimental results clearly show that our proposed model outperforms the state-of-the-art benchmark approaches.

1 Introduction

With the prevalence of Location Based Social Networks (LBSNs), personalized merchant recommendation has become very popular and attracted much attention from industry and academia. Personalized merchant recommendation not only satisfies users personalized preferences for visiting new merchants, such as restaurants, stores and theatres, but also increases merchants revenues. As customers can easily make their evaluation by rating and express their opinions freely by writing their reviews on merchants, mining user real preferences on

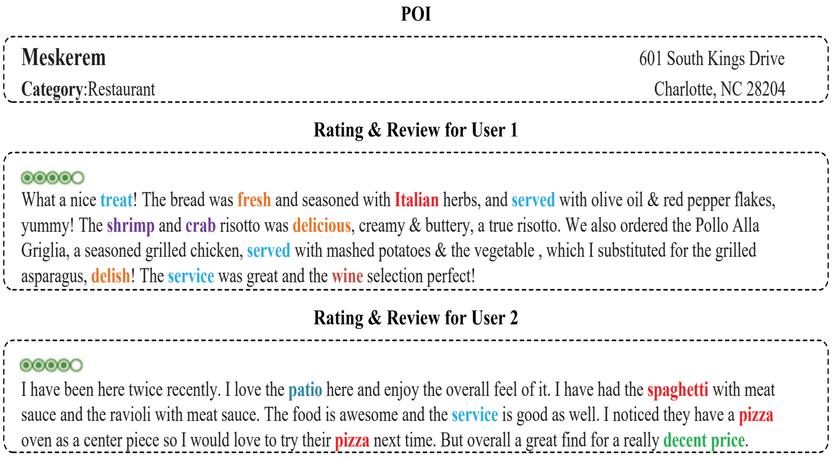

Fig. 1. Example of Yelp rating and review.

various topics from user reviews will be useful for personalized merchant recommendation. To solve the data sparsity and the cold start problem in real recommendation system, some methods further incorporate contextual information, such as social networks, temporal information and geographical location to alleviate the cold start problem [3,10,24].

Though current techniques achieve great progress to alleviate these problems, it is still very difficult to learn user preferences across various topics from user reviews. As we know, review texts may consist of many topics. From all the reviews written by a specific user, we can infer what topics the specific user gives more attention to. However, users with same rating on same merchant may have different preferences on different topics. Let us consider a real restaurant example shown in Fig.1 where two users have same rating on same restaurant. From the perspective of the user, we can infer that user 1 pays more attention to the *flavour of the food* and the *service*, while user 2 pays more attention to the *price* and the *service*. From the perspective of the merchant, we can conclude that this restaurant may be an Italian restaurant, which offers seafood with a good *service* and charges a decent *price*. So we can come to the conclusion that the rating reflects user general evaluation on the merchant, while the review reflects user preferences across many topics. Therefore, both the rating and the review should be integrated for merchant recommendation.

In this paper, we propose an integrated model by fusing user ratings and user real preferences on various topics from user reviews in a unified framework, named "Collaborative Filtering meets Reviews" (CFMR), which combines Collaborative Filtering technique and Topic Modeling method based on a graphical model. We assume that the overall rating consists of two parts. One part is the collaborative rating, which is based on the rating behavior of users in the past. The other is the latent rating, which is based on the current user's preference.

The experimental results on two real-world datasets show that our proposed model outperforms the state-of-the-art benchmark approaches in terms of effectiveness and efficiency.

The structure of the remainder of this paper is as follows. In Section 2 we will give an overview of the related work. In Section 3 we will describe the problem definition. Then, in Section 4, we will detail our new model in the form of graphical model and the concrete algorithm for merchant recommendation based on our model. In Section 5 we will present and discuss the experimental result on two datasets. The settings of the parameters is also included in this section. Finally, we will conclude our main contributions and discuss the future work in Section 6.

2 Related Work

In this section, we will review a number of existing works on recommender systems.

Recommender systems can be generally classified into two groups: Content-based Filtering [20] and Collaborative Filtering [8]. The latent factor model based on matrix factorization is one of the typical models in Collaborative Filtering, such as basic Matrix Factorization [14] and Probabilistic Matrix Factorization [21]. In the recent years, some additional information, such as item taxonomy information [23] and social networks [17], have been incorporated to improve the prediction quality. In [6], the authors presented the performance of matrix factorization algorithms and memory based models on the task of recommending long tail items.

With the prevalence of mobile devices and Web 2.0 technologies, location-based social networks (LBSNs) allow users to share their experiences and opinions on merchants. There are a lot of applications, such as user behavior study [25] and online retail store placement [12] based on Points-of- Interest (POI) recommendation. In recent years, many additional information have been explored to improve the recommendation quality for POI recommendation in LBSNs, such as geographical location [4] and social connections [5]. Our approach differs from previous methods in that we try to learn user preferences by mining the reviews.

As review texts contain very useful information to learn user preferences, there are a few efforts to combine this information to make predictions for ratings. In the work [9], the authors created bag of words from the top frequent words in all reviews and proposed the combination of Linear Regression model. The authors of [11] proposed the decomposition of hidden topics over all reviews and predicted rating over every hidden topic. In [19], the authors proposed a generic rating prediction model by incorporating user preferences in sentiment rating prediction to improve the correlation with ground ratings. The most related study to our work is the work of [16] where the boot-strapping aspect segmentation algorithm proposed in [22] is directly used to define the meaningful aspects. However, the aspect representation based on the boot-strapping aspect segmentation algorithm proposed in [22] needs to provide pre-specified aspect

Table 1. Mathematical notations.

Symbol	Size	Description
D	N	all the users
P	M	all the merchants
R	$N \times M$	overall rating matrix
R^c	$N \times M$	collaborative rating matrix
R^d	$N \times M$	latent rating matrix
U	$N \times L$	user latent factor matrix
V	$M \times L$	merchant latent factor matrix
d_j^i	$-$	review on merchant j written by user i
T	$(.)$	multiple topics from user reviews
K	$-$	the number of keywords in each topic representation
W_j	$K \times \|T\|$	word frequency matrix for merchant j
Z_j	$1 \times \|T\|$	topic ratings vector for merchant j
γ	$K \times \|T\|$	word sentiment polarity matrix
β_i	$1 \times \|T\|$	preference vector for user i
$\sigma, \sigma_U, \sigma_V, \sigma_\beta$	R	variance of the priors

keywords by users. The major difference between our work and these review based methods is that, we proposed a graphical model by fusing Collaborative Filtering technique and Topic Modeling approach to learn user preferences without users' specification of aspect keywords.

3 Notation and Definition

In this section, we introduce the relevant notations and the formal definitions. The input is a set of ratings of some merchants, where each rating has a review. All the notations we used in this paper are shown in Table 1.

Formally, let $D = \{\mu_1, \ldots, \mu_N\}$ be the set of users. let $P = \{\upsilon_1, \ldots, \upsilon_M\}$ be the set of merchants. A rating R_{ij} user μ_i gives to merchant υ_j indicates the preference the user μ_i shows for the merchant υ_j, where high rating means the high preference. Here, ratings are integers ranging from 1 (star) indicating few interest to 5 (star) indicating strong interest. The rating R_{ij}^c indicates the collaborative rating which is achieved by general Collaborative Filtering method. The rating R_{ij}^d indicates the latent rating which is mined in the reviews. The notation d_j^i represents the review written by user μ_i for merchant υ_j, which implicitly indicates the user preferences and the merchant quality. We aggregate all the reviews written for the merchant υ_j, i.e., $\mathbf{d}_j = \{d_j^1, \ldots, d_j^i, \ldots, d_j^N\}$, to analyze the quality of the merchant υ_j. In order to understand our proposed model, we introduce the following definitions.

Definition (Topics) The review is made up of many topics. Users may pay more attention to some topics. The topics are denoted as $\mathbf{T} = \{T^1, \ldots, T^i, \ldots, T^{|\mathbf{T}|}\}$. The topics can be *price*, *food* and *service* in the case of a restaurant.

Definition (Topic Representation) Every topic $\mathbf{T}^i$ consists of K most representative keywords, i.e., $\mathbf{T}^i = \{w_1^i, w_2^i, \dots, w_K^i\}$. Each topic keyword is selected from the review corpus.

Definition (Topic Ratings) Topic ratings is a $|\boldsymbol{T}|$ dimensional vector of score over all topics, denoted as $\boldsymbol{Z}_j = \{Z_j^1, \dots, Z_j^k, \dots, Z_j^{|\boldsymbol{T}|}\}$. Topic ratings indicate the quality offered by the merchant on topics. A high topic rating means the merchant does well on this topic.

Definition (User Preferences) User preferences is a $|\boldsymbol{T}|$ dimensional vector of weights over all topics, where the k-th dimension indicates the degree of user preference toward the topic T^k, denoted as $\boldsymbol{\beta}_i = \{\beta_i^1, \dots, \beta_i^k, \dots, \beta_i^{|T|}\}$. A high weight means more attention is paid to the corresponding topic.

4 Model Specification

In this section, we first introduce the topic representation and the calculation of the topic ratings. Then, we introduce the proposed model, titled "Collaborative Filtering meets Reviews" (CFMR).

4.1 Topic Representation and Calculation of the Topic Ratings

A major challenge in our work is defining meaningful topics. Because we do not have the supervision information about how many topics we should define and what each topic is made up of. In this paper, we apply Latent Dirichlet Allocation (LDA) [2] model to find the underlying topics and components of the corresponding topic. We train the LDA model on the review corpus made up of all the reviews from all users. For each topic, there exists a word distribution. Every word is associated with a probability value which represents the percentage that this word makes up of the corresponding topic. We first sort the words by the probability value descendingly for each topic. Then we select K most representative words with high probability as our topic keywords.

The calculation of the topic ratings is as follows. Suppose that we have $|T|$ topics and each topic is represented by K keywords. We get a $K \times |T|$ keyword matrix. We aggregate all the reviews written for merchant v_j, i.e., $\mathbf{d}_j = \{d_j^1, \dots, d_j^i, \dots, d_j^N\}$. Based on the keyword matrix, we can map the aggregated review $\mathbf{d}_j$ to a word frequency matrix W_j in which each column corresponds to a keyword frequency vector in terms of this topic representation. Each element in the matrix W_j represents the frequency of the word in the aggregated review $\mathbf{d}_j$. In natural language processing, each word is associated with a negative, neutral or positive sentiment which shows the level of reviewer preferences. So we introduce a sentiment polarity parameter matrix γ which can be learned by minimizing the objective function given by Eq.10. Each topic rating is the

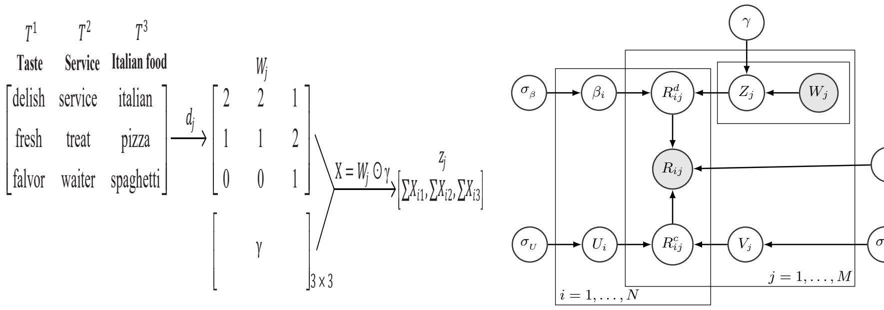

Fig. 2. Left panel: topic representation and calculation of the topic ratings. Right panel: the graphical model for the CFMR model.

dot product of the keyword frequency vector in terms of this topic representation and the corresponding sentiment polarity vector. The Hadamard product of matrix W_j and matrix γ is denoted as matrix X, i.e.

$$X = W_j \odot \gamma, \tag{1}$$

where $\odot$ is the Hadamard product. Then we can calculate the topic ratings Z_j by doing the column sum on matrix X, i.e.

$$Z_j = \left[\sum X_{i1}, \sum X_{i2}, \ldots, \sum X_{i|T|} \right]. \tag{2}$$

Figure 2 (left panel) shows a simple example where we assume the $\mathbf{d}_j$ consists of two reviews shown in Fig.1. We only present three topics and show three keywords in each topic due to space limitation.

4.2 Collaborative Filtering meets Reviews

In this section, we apply an integrated model, titled "Collaborative Filtering meets Reviews" (CFMR), which combines the collaborative rating and the latent rating to approximate the overall rating.

The Generation Assumption We assume that the overall rating consists of two parts. One part is the collaborative rating, which is based on the rating behavior of users in the past. The other is the latent rating, which is based on the current user's preference. We further assume that the latent rating is a weight sum of all the topic ratings, where the weights show the relative preference the user placed on each topic. The graphical model for the CFMR model is shown in Fig.2 (right panel).

The CFMR Model As aforementioned, the overall rating consists of the collaborative rating and the latent rating. We adopt a probabilistic model with Gaussian observation noise (see Fig.2 (right panel)). We define the condition distribution over the observed ratings as

$$p(R|R^c, R^d, \sigma^2) = \prod_{i=1}^{N} \prod_{j=1}^{M} \left[\mathcal{N}(R_{ij}|R_{ij}^c + R_{ij}^d, \sigma^2) \right]^{I_{ij}}, \tag{3}$$

where $\mathcal{N}(x|\mu, \sigma^2)$ represents the probability density function of the Gaussian distribution with mean μ and variance σ^2, and I_{ij} is the indicator function that is equal to 1 if user i rated merchant j and equal to 0 otherwise.

The collaborative rating R_{ij}^c can be calculated as follows based on matrix factorization [14]:

$$R_{ij}^c = U_i V_j^T, \tag{4}$$

where the row vector U_i and V_j represent the user feature vector and the merchant feature vector respectively. We place zero-mean spherical Gaussian priors [7,1] on user and merchant feature vectors as follows:

$$p(U|\sigma_U^2) = \prod_{i=1}^{N} \mathcal{N}(U_i|0, \sigma_U^2 \boldsymbol{I}), \qquad p(V|\sigma_V^2) = \prod_{j=1}^{M} \mathcal{N}(V_j|0, \sigma_V^2 \boldsymbol{I}). \tag{5}$$

As we assume that the latent rating is a weight sum of all topic ratings, where the weights show the relative preference the user placed on each topic, so we define the latent rating R_{ij}^d as

$$R_{ij}^d = \beta_i Z_j^T, \tag{6}$$

where the row vector β_i and Z_j represent the preference vector of user i and the topic ratings vector of merchant j respectively.

Note that the parameter matrix γ characterizes the sentiment polarity of all the keywords. In order to simplify the discussion, we place a uniform distribution on parameter matrix γ. As we show in Section 4.1, topic ratings vector Z_j depends on the parameter γ and the word frequency matrix W_j. So we do not place any priors on topic ratings vector Z_j. We just place zero-mean spherical Gaussian priors [7,1] on user preference vectors as follows:

$$p(\beta|\sigma_\beta^2) = \prod_{i=1}^{N} \mathcal{N}(\beta_i|0, \sigma_\beta^2 \boldsymbol{I}). \tag{7}$$

From Eq.3,4,6, we can get the following equation, i.e.

$$p(R|U, V, \beta, \sigma^2) = \prod_{i=1}^{N} \prod_{j=1}^{M} \left[\mathcal{N}(R_{ij}|U_i V_j^T + \beta_i Z_j^T, \sigma^2) \right]^{I_{ij}}. \tag{8}$$

For the simplification of the equation, let $\Omega = \{U, V, \beta, \gamma\}$ and $\Theta = \{\sigma, \sigma_U, \sigma_V, \sigma_\beta\}$. The log of the posterior distribution over user feature, merchant feature and user preference vectors is given by

$$\begin{aligned}\ln p(\Omega|R,\Theta) = & \ln p(R|U,V,\beta,\gamma,\sigma^2) + \ln p(U|\sigma_U^2) + \ln p(V|\sigma_V^2) + \\ & \ln p(\beta|\sigma_\beta^2) + \ln p(\gamma) - \ln p(R) \\ = & -\frac{1}{2\sigma^2}\sum_{i=1}^{N}\sum_{j=1}^{M}\boldsymbol{I}_{ij}(R_{ij} - U_i V_j^T - \beta_i Z_j^T)^2 - \frac{1}{2\sigma_U^2}\sum_{i=1}^{N} U_i U_i^T - \\ & \frac{1}{2\sigma_V^2}\sum_{j=1}^{M} V_j V_j^T - \frac{1}{2\sigma_\beta^2}\sum_{i=1}^{N}\beta_i\beta_i^T - \frac{1}{2}\left(\sum_{i=1}^{N}\sum_{j=1}^{M}\boldsymbol{I}_{ij}\right) ln\sigma^2 - \\ & \frac{1}{2}\left(LNln\sigma_U^2 + LMln\sigma_V^2 + |T|Nln\sigma_\beta^2\right) + C, \end{aligned} \tag{9}$$

where L represents the length of the user feature vector and C represents a constant that does not depend on the parameters. Maximizing the log-posterior in Eq.9 with hyperparameters ($\sigma^2, \sigma_U^2, \sigma_V^2, \sigma_\beta^2$) kept fixed is equivalent to minimizing the sum-of-squared-errors objective function with quadratic regularization terms:

$$\begin{aligned}L = & \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{M}\boldsymbol{I}_{ij}(R_{ij} - U_i V_j^T - \beta_i Z_j^T)^2 + \frac{\lambda_U}{2}\sum_{i=1}^{N}\|U_i\|_{Fro}^2 + \\ & \frac{\lambda_V}{2}\sum_{j=1}^{M}\|V_j\|_{Fro}^2 + \frac{\lambda_\beta}{2}\sum_{i=1}^{N}\|\beta_j\|_{Fro}^2, \end{aligned} \tag{10}$$

where $\lambda_U = \sigma^2/\sigma_U^2$, $\lambda_V = \sigma^2/\sigma_V^2$, $\lambda_\beta = \sigma^2/\sigma_\beta^2$, and $\|.\|_{Fro}^2$ denotes the Frobenius Norm. Note that topic ratings vector Z_j depends on the parameter matrix γ. A local minimum of the objective function given by Eq.10 can be calculated by performing the gradient descent in U, V, β, γ by moving in the opposite direction of the gradient. The gradient of the U, V, β, γ is as followed:

$$\begin{aligned}\frac{\partial L}{\partial U_i} &= \sum_j \boldsymbol{I}_{ij}[(U_i V_j^T + \beta_i Z_j^T) - R_{ij}]V_j + \lambda_U U_i \\ \frac{\partial L}{\partial V_i} &= \sum_i \boldsymbol{I}_{ij}[(U_i V_j^T + \beta_i Z_j^T) - R_{ij}]U_i + \lambda_V V_j \\ \frac{\partial L}{\partial \beta_i} &= \sum_j \boldsymbol{I}_{ij}[(U_i V_j^T + \beta_i Z_j^T) - R_{ij}]V_j + \lambda_\beta \beta_i \\ \frac{\partial L}{\partial \gamma_k} &= \sum_{i,j} \boldsymbol{I}_{ij}[(U_i V_j^T + \beta_i Z_j^T) - R_{ij}]\beta_i^k W_j^{:,k} \end{aligned} \tag{11}$$

where γ_k is a word sentiment polarity vector for k-th topic.

Algorithm 1: Merchant recommendation.

Input : Reviews D , Rating Matrix R
Output: $\Omega = \{U, V, \beta, \gamma\}$

Random initialize Ω;
for $step = 1$ **to** $MAXSTEP$ **do**
    for $i \leftarrow 0$ **to** N **do**
        $U_i \leftarrow U_i - \eta \mathbf{\nabla}_{U_i}$;
        $\beta_i \leftarrow \beta_i - \eta \mathbf{\nabla}_{\beta_i}$;
    for $j \leftarrow 0$ **to** M **do**
        $V_j \leftarrow V_j - \eta \mathbf{\nabla}_{V_j}$;
        $\gamma_j \leftarrow \gamma_j - \eta \mathbf{\nabla}_{\gamma_j}$;
    if *converge* **then**
        stop iteration;
    change the learning rate η *as step increases*;
for *each user* i **do**
    for *each merchant* j **do**
        calculate the rating R_{ij};
    recommend merchants with high rating for user i;

Merchant Recommendation by CFMR Model Merchant recommendation can be implemented by selecting merchants with high ratings. The algorithm of merchant recommendation is detailed in Algorithm 1.

Complexity Analysis The review preprocessing and the training process of LDA model only cost a constant time, which is not relevant to complexity of our algorithm. The main complexity of our algorithm is evaluating the objective function in Eq.10 and its gradient against variables in Eq.11. Owing to the spartity of matrix R, the computational complexity of the objective function is $O(\rho(F + |T|))$, where ρ is the number of the nonzero entries in matrix R, F is the dimension of the row vector U_i and the $|T|$ is the dimension of the row vector β_i. The computaional complexity of gradients is also $O(\rho(F + |T|))$. Therefore, the computational time of our algorithm is linear with repect to the number of observations in the matrix R.

5 Experimental Results

In this section, we compare our CFMR model with some baseline methods in terms of effectiveness and efficiency.

5.1 Data Set and Setting

We perform our experiments on two real-world datasets, Yelp dataset[1] and TripAdvisor dataset[2]. The two datasets offer the ratings and the corresponding

[1] https://www.yelp.com/dataset_challenge
[2] http://times.cs.uiuc.edu/~wang296/Data/

reviews, making them ideal datasets for our model. For Yelp dataset, we choose the restaurant category which covers more than 1/3 of the total reviews and filter out users and merchants with less than 20 records. For TripAdvisor dataset, we filter out users and merchants with less than 10 records. The final Yelp dataset consists of 6335 users, 4096 merchants and 220454 reviews. The final TripAdvisor dataset consists of 13930 users, 4489 merchants and 216929 reviews.

We perform some necessary preprocessing on the reviews before Topic Modeling: (i) transform the words to lower cases; (ii) filter out the stop words; (iii) stem each word in the review corpus to its root.

In the experiments, we try various values for the learning rate and experiment with various values of L (the length of the user feature vector U_i), finally we chose to use a learning rate of 0.005 and the value of L is set to 30. We also investigate the impact of the number keywords in each topic representation and the number of topics, the number keywords in each topic representation is fixed to 10 (i.e., $K = 10$) and the number of topics is fixted to 20 (i.e., $|T| = 15$).

5.2 Baseline Methods

We compare our method with the following four baseline models:

- **PMF** : Probabilistic Matrix Factorization is proposed in [21] by modeling user preference matrix as a product of two lower-rank user and merchant matrices.
- **LDAMF** : This method is proposed in [18] which utilizes the information buried in the review texts by fitting an LDA model on the review corpus and then treating the learned topic distribution on merchants (or users) as the latent factors in Matrix Factorization.
- **SVD++** : This is the state-of-the-art method [13] whose author had won the Netflix Prize. This approach is based on a matrix factorization model by incorporating implicit feedback information.
- **GM-L2** : This method is proposed in [16] which follows the boot-strapping aspect segmentation algorithm proposed in [22] to achieve the aspect representation.

5.3 Accuracy Prediction

Evaluation Metrics We use two typical accuracy metrics, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), to measure the performance by comparing the observed rating against the predicted rating. The definitions are as follows:

$$MAE = \frac{1}{|L|}\sum_{i,j}|R_{i,j} - \widehat{R}_{i,j}|, \quad RMSE = \sqrt{\frac{1}{|L|}\sum_{i,j}(R_{i,j} - \widehat{R}_{i,j})^2},$$

where $R_{i,j}$ and $\widehat{R}_{i,j}$ indicate the observed rating and the predicted rating respectively, and $|L|$ is the number of all test cases.

Table 2. Comparison on accuracy prediction.

Datasets	Ratio	Metric	LDAMF	PMF	SVD++	GM-L2	CFMR
Yelp	90%	MAE	0.8211	0.7957	0.7733	0.7544	0.7464
		improved	10.01%	6.61%	3.60%	1.07%	
		RMSE	1.0592	1.0224	1.0019	0.9787	0.9684
		improved	9.38%	6.19%	3.46%	1.06%	
	70%	MAE	0.8254	0.7938	0.7774	0.7573	0.7497
		improved	10.10%	5.88%	3.69%	1.01%	
		RMSE	1.0662	1.0266	1.0071	0.9823	0.9727
		improved	9.61%	5.54%	3.54%	0.99%	
TripAdvisor	90%	MAE	0.7946	0.7629	0.7431	0.7259	0.7154
		improved	11.07%	6.64%	3.87%	1.47%	
		RMSE	1.0255	0.9821	0.9562	0.9349	0.9198
		improved	11.49%	6.77%	3.96%	1.64%	
	70%	MAE	0.8003	0.7707	0.7460	0.7308	0.7192
		improved	11.27%	7.16%	3.73%	1.61%	
		RMSE	1.0332	0.9928	0.9600	0.9431	0.9232
		improved	11.92%	7.54%	3.99%	2.16%	

Experimental Results For Yelp dataset, the regularization parameters are set to $\lambda_U = \lambda_V = \lambda_\beta = 0.003$. For TripAdvisor dataset, the regularization parameters are set to $\lambda_U = \lambda_V = \lambda_\beta = 0.001$. We use different ratio of training data to test all algorithms and the results are shown in Table 2. Compared with the baseline methods, our model achieves better performance on accuracy prediction. The reason is that our model incorporates the review information and takes user preferences across multiple topics into consideration. Our approach also performs better than the recently proposed method GM-L2 [16] because the number of topics we consider is more than the number of aspects used in GM-L2 [16]. We also find that the more data used to training, the more accurate prediction results we got. The comparison results on TripAdvisor dataset are similar to the results on Yelp dataset.

5.4 Cold Start Problem

Experimental Configuration We randomly split the total dataset into two parts: training set and testing set according to the split ratio of 20 : 80. For any test user, we randomly select 20% of user total rating records as the observed merchant ratings and the remaining as the held-out merchant ratings. Our main objective is to use the ratings of the observed merchants to predict the ratings of the held-out merchants. In order to measure the performance of our model on the cold start problem, we divide the test users into groups according to the number of their observed ratings in the training set, i.e., 1-3, 4-5, 6-7, 8-9, 10-11, 12-13 on Yelp dataset.

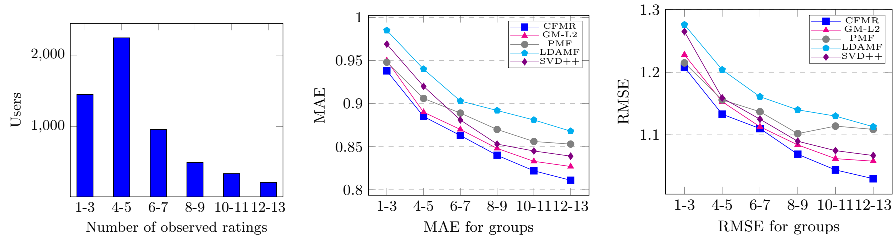

Fig. 3. Yelp: Left panel: distribution of users in the training dataset. Middle panel: MAE for groups. Right panel: RMSE for groups.

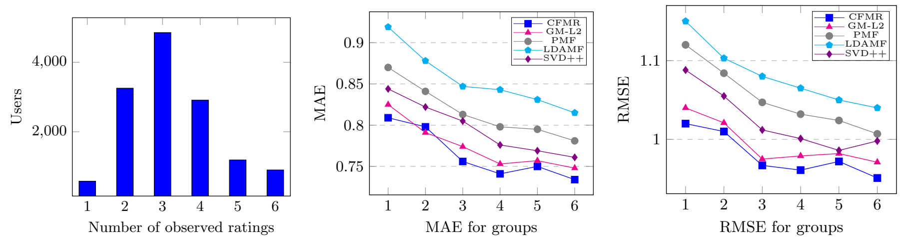

Fig. 4. TripAdvisor: Left panel: distribution of users in the training dataset. Middle panel: MAE for groups. Right panel: RMSE for groups.

Experimental Results For Yelp dataset, the regularization parameters are set to $\lambda_U = \lambda_V = \lambda_\beta = 0.005$. For TripAdvisor dataset, the regularization parameters are set to $\lambda_U = \lambda_V = \lambda_\beta = 0.002$. Figure 3 (left panel) shows the distribution of the observed ratings in the training set according to the split ratio of 20:80 on Yelp dataset. Figure 3 (left panel) and Fig 3 (right panel) report the MAE and RMSE results on Yelp dataset respectively. We notice that the group where users have more ratings achieves better accuracy than other groups. We can also observe that our approach outperforms the other baseline methods because our model can learn user preferences across multiple topics. LDAMF method behaves worst among these methods, since it only incorporates the review information. We also present the performance of all algorithms on TripAdvisor dataset in Fig.4. From Fig.4, we can see that the experimental results of all algorithms on TripAdvisor dataset are not as well as the performance on Yelp dataset because TripAdvisor dataset is more sparser than Yelp dataset.

5.5 Long Tail Effect

Evaluation Metrics We utilize the recall measurement to evaluate the performance of our model to discover long tail merchant (less popular merchant). This testing methodology adopted in this paper is applied in [11], which has been widely used to evaluate the recommender system. The detailed procedure about how this experiment is conducted is as followed. For each dataset, known ratings

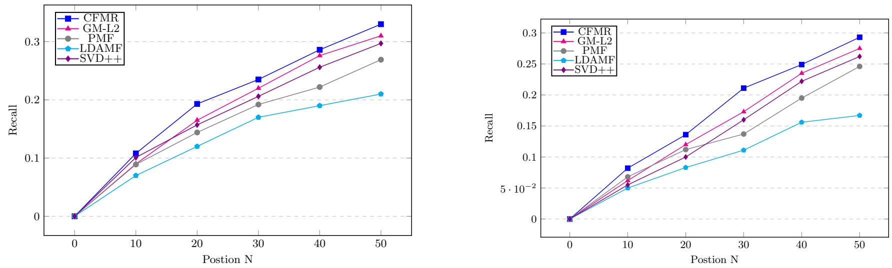

Fig. 5. Left panel: recall on Yelp. Right panel: recall on TripAdvisor.

are split into two subsets: training set and testing set. The testing set contains only 5-stars ratings. For each user, we randomly select one long tail rating with 5-stars which will be added into the testing set. The remaining ratings are as the training set. As expected, the testing set is not used for training. In order to calculate recall, we first train the model on the training set. Then for each long tail merchant i rated 5-stars by user u in the testing set: (i) randomly select 800 additional merchants unrated by user u. (ii) predict the ratings for the merchant i as well as the additional 800 merchants. (iii) form a ranked list by order in all the 801 merchants based on their ratings. (iv) form a top-N recommendation list by selecting the top-N ranked merchants from the ranked list. If the merchant i is in the top-N recommendation list, we have a hit ($hit = 1$). Otherwise we have a miss ($hit = 0$).

The recall is defined by averaging over all test cases:

$$Recall = \frac{\sum hit}{|Test|},$$

where $|Test|$ is the number of all test cases.

Experimental Results For Yelp dataset, the regularization parameters are set to $\lambda_U = \lambda_V = \lambda_\beta = 0.008$. For TripAdvisor dataset, the regularization parameters are set to $\lambda_U = \lambda_V = \lambda_\beta = 0.001$. We present the performance on top-N in the range from 0 to 50, since the larger N is meaningless in the typical top-N recommendation task. Figure 5 (left panel) reports the performance of recommendation algorithms on Yelp dataset. Clearly, the models achieve different performance in terms of top-N recommendation. The recall of CFMR at $N = 30$ is 0.225, i.e., the model can place a long tail merchant in the top-30 with the probability of 22.5%. From Fig.5 (left panel), we can see that our model is better than the model GM-L2 [16] because the number of topics we considered is more than the number of aspects in GM-L2 [16]. We can also find that the model of SVD++ presents better performance than the method of PMF because the SVD++ model incorporates implicit feedback information. Figure 5 (right panel) reports the performance of all algorithms on TripAdvisor dataset. It is apparent that the trend of the comparison result is similar to that of Fig.5 (left panel).

6 Conclusion

In this paper, we propose an integrated graphical model named CFMR by incorporating user real preferences on various topics from user reviews into Collaborative Filtering technique for personalized merchant recommendation. We assume that the overall rating consists of two parts. One part is the collaborative rating, which is based on the rating behavior of users in the past. The other is the latent rating, which is based on the current user's preference. Our work mainly focuses on the topic representation and the calculation of the latent rating. We conducted extensive experiments on two real-world datasets and the experimental results demonstrate that our model outperforms other methods in terms of effectiveness and efficiency. In future, we would like to consider the impact of the additional information, such as geographic information [4] and social information [15], to further improve the performance of merchant recommendation.

Acknowledgments. This research was partially supported by grants from the National Natural Science Foundation of China (NSFC, Grant No.U1605251), the National Science Foundation for Distinguished Young Scholars of China (Grant No.61325010), and the NSFC Major research program (Grant No.91546103).

References

1. C. M. Bishop. Probabilistic principal component analysis. 1997.
2. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.
3. A. J. Chaney, D. M. Blei, and T. Eliassi-Rad. A probabilistic model for using social networks in personalized item recommendation. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 43–50. ACM, 2015.
4. C. Cheng, H. Yang, I. King, and M. R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *Aaai*, volume 12, pages 17–23, 2012.
5. E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090. ACM, 2011.
6. P. Cremonesi, Y. Koren, and R. Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 39–46. ACM, 2010.
7. D. Dueck, B. Frey, D. Dueck, and B. J. Frey. Probabilistic sparse matrix factorization. *University of Toronto technical report PSI-2004-23*, 2004.
8. M. D. Ekstrand, J. T. Riedl, and J. A. Konstan. Collaborative filtering recommender systems. *Foundations and Trends in Human-Computer Interaction*, 4(2):81–173, 2011.
9. M. Fan and M. Khademi. Predicting a business star in yelp from its reviews text alone. *arXiv preprint arXiv:1401.0864*, 2014.
10. H. Gao, J. Tang, X. Hu, and H. Liu. Exploring temporal effects for location recommendation on location-based social networks. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 93–100. ACM, 2013.

11. J. Huang, S. Rogers, and E. Joo. Improving restaurants by extracting subtopics from yelp reviews. *iConference 2014 (Social Media Expo)*, 2014.
12. D. Karamshuk, A. Noulas, S. Scellato, V. Nicosia, and C. Mascolo. Geo-spotting: mining online location-based services for optimal retail store placement. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 793–801. ACM, 2013.
13. Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.
14. Y. Koren, R. Bell, C. Volinsky, et al. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
15. H. Li, Y. Ge, and H. Zhu. Point-of-interest recommendations: Learning potential check-ins from friends. In *Proceedings of the 22th ACM SIGKDD international conference on on Knowledge discovery and data mining. ACM*, 2016.
16. X. Li, G. Xu, E. Chen, and L. Li. Learning user preferences across multiple aspects for merchant recommendation. In *Data Mining (ICDM), 2015 IEEE International Conference on*, pages 865–870. IEEE, 2015.
17. H. Ma, I. King, and M. R. Lyu. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 203–210. ACM, 2009.
18. J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172. ACM, 2013.
19. S. Mukherjee, G. Basu, and S. Joshi. Incorporating author preference in sentiment rating prediction of reviews. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 47–48. ACM, 2013.
20. M. J. Pazzani and D. Billsus. Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer, 2007.
21. R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS*, volume 20, pages 1–8, 2011.
22. H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 783–792. ACM, 2010.
23. L.-T. Weng, Y. Xu, Y. Li, and R. Nayak. Exploiting item taxonomy for solving cold-start problem in recommendation making. In *2008 20th IEEE International Conference on Tools with Artificial Intelligence*, volume 2, pages 113–120. IEEE, 2008.
24. Q. Yuan, G. Cong, and A. Sun. Graph-based point-of-interest recommendation with geographical and temporal influences. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 659–668. ACM, 2014.
25. V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang. Collaborative filtering meets mobile recommendation: A user-centered approach. In *AAAI*, volume 10, pages 236–241, 2010.