Expert Systems With Applications 131 (2019) 1–8

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Bayesian probabilistic tensor factorization for recommendation and rating aggregation with multicriteria evaluation data

Hiroki Morise[a], Satoshi Oyama[a,b,c,*], Masahito Kurihara[a]

[a] Graduate School of Information Science and Technology, Hokkaido University, Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido 060 - 0814, Japan
[b] Global Institution for Collaborative Research and Education, Hokkaido University, Kita 8, Nishi 5, Kita-ku, Sapporo, Hokkaido 060 - 0808, Japan
[c] RIKEN Center for Advanced Intelligence Project, Nihonbashi 1-chome Mitsui Building, 15th floor, 1-4-1 Nihonbashi, Chuo-ku, Tokyo 103 - 0027, Japan

ARTICLE INFO

Article history:
Received 8 August 2018
Revised 18 April 2019
Accepted 18 April 2019
Available online 19 April 2019

Keywords:
Recommendation
Multi-criteria rating
Collaborative filtering
Rating aggregation
Bayesian probabilistic models

ABSTRACT

Ratings by users on various items such as products and services have become easily available on the Web. Also available in many cases, in addition to an overall rating for each item by each user, are multicriteria ratings from different viewpoints. Our previous study showed that multicriteria rating approaches performed better than single-criterion ones for both recommendation and rating aggregation. We have now formulated a Bayesian probabilistic model for multicriteria evaluation as an alternative to low-rank approximation. We evaluated the performance of this model, in which model capacity is controlled by integrating over all model parameters, and investigated whether it can be made to work more efficiently by using a Markov chain Monte Carlo method for both recommendation and rating aggregation. It performed better than low-rank approximation methods that obtain a maximum a posteriori estimate by fitting to the data.

1. Introduction

Ratings by users on various items such as products and services are now easily available on the Web. However, it is difficult to acquire reliable information because of information overload—there are usually many ratings for each item on the Web. Recommendation techniques in particular help consumers avoid information overload and find interesting items. One of the most promising types of recommendation methods is collaborative filtering (CF). Matrix factorization, which uses a matrix of users and items, is used in one of the most commonly used approaches to CF and has been shown to be better than conventional collaborative filtering for product recommendation (Koren, Bell, & Volinsky, 2009). It has thus been attracting much attention.

In many cases, in addition to an overall rating for each item by each user, multicriteria ratings from different viewpoints are also available. We previously investigated the effectiveness of existing CF methods for large-scale sparse multicriteria rating data (Morise, Oyama, & Kurihara, 2017). We formulated rating aggregation as a CF problem and applied several methods to it. The multicriteria rating approaches performed better than the single-criterion ones, and the CF methods using a multicriteria rating predicted aggregated ratings more accurately than ones using a single-criterion rating. Specifically, among low-rank approximation methods, the tensor factorization method consistently performed better than the matrix factorization methods.

However, low-rank approximation methods tend to overfit the data unless the regularization parameters are carefully adjusted. We have developed a Bayesian probabilistic model for multicriteria evaluation that uses a low-rank tensor factorization recommendation model. We evaluated this model, in which model capacity is controlled by integrating over all parameters, as an alternative to low-rank approximation methods and investigated whether it can be made to work more efficiently by using a Markov chain Monte Carlo method for both recommendation and rating aggregation.

This paper is organized as follow. Section 2 describes matrix factorization and tensor factorization recommendation methods. In Section 3, we introduce existing Bayesian probabilistic models for recommendation. In Section 4, we present our Bayesian probabilistic model that uses multicriteria evaluation data. In Section 5, we describe our experiment of its accuracy for both recommendation and rating aggregation. In Section 6, we summarize the key points and mention future work.

The main contributions of this work are summarized below:

- To the best of our knowledge, our work is the first attempt to apply Bayesian probabilistic tensor factorization to multicriteria recommendation. Our model, which we call "Bayesian probabilistic tensor factorization for multicriteria (BPTF-MC),"

[*] Corresponding author.
E-mail addresses: morise.hiroki@complex.ist.hokudai.ac.jp (H. Morise), oyama@ist.hokudai.ac.jp (S. Oyama), kurihara@ist.hokudai.ac.jp (M. Kurihara).

https://doi.org/10.1016/j.eswa.2019.04.044
0957-4174/

predicts the overall rating and the rating from each viewpoint simultaneously. It does this by using multicriteria latent features as additional factors.

- The BPTF-MC model enables the prediction of ratings for items by each user and of aggregated ratings from the evaluations of a small number of users.
- Experimental results for the Rakuten public datasets show that the BPTF-MC model achieves better performance than single-criterion models and low-rank tensor factorization models for both recommendation and rating aggregation.

2. Collaborative filtering by low-rank approximation methods

In this section, we describe matrix factorization and tensor factorization recommendation methods that predict an unknown evaluation rating for each item for each user from the ratings.

2.1. Matrix factorization

One of the most commonly used approaches to CF is based on the matrix factorization model (Koren et al., 2009). This approach characterizes both users and items by using latent factors from ratings. The matrix contains the evaluation rating for each item by each user. For example, for I users and J items, given $I \times J$ user-item rating matrix $R = [R_{ij}]_{I \times J}$, the matrix factorization model represents rating matrix R as the product of K-rank factors $R \approx U^T V$, where $U \in R^{K \times I}$ and $V \in R^{K \times J}$. K are the number of latent factors for users and items. In general, K is smaller than I and J.

The latent representations of the users and items are computed by minimizing the following regularized squared error from observed ratings:

$$\min_{U,V} \sum_{(i,j) \in d} (R_{ij} - U_i^T V_j)^2 + \lambda(||U_i||^2 + ||V_j||^2), \tag{1}$$

where d is the set of observed user and item pairs of R, and constant λ works to avoid overfitting the observed evaluation ratings.

The problem of some users giving prejudiced ratings and the problem of some items being evaluated on the basis of those ratings are avoided by adding biases.

The latent representations of the users and items are computed by minimizing the following regularized squared error from the observed ratings:

$$\min_{U,V} \sum_{(i,j) \in d} (R_{ij} - U_i^T V_j - \mu - b_i - b_j)^2$$
$$+ \lambda(||U_i||^2 + ||V_j||^2 + b_i^2 + b_j^2). \tag{2}$$

The overall average rating is defined by μ; parameters b_i and b_j represent the observed variations from the averages for users and items.

2.2. Tensor factorization

In matrix factorization, the relationship between two objects is modeled using a low- rank matrix. However, sometimes there are established relationships among more than two objects. These relationships can be represented as a multidimensional array, which is a generalization of matrix factorization (Xiong, Chen, Huang, Schneider, & Carbonell, 2010). In our experiment, we used the relationships among I users U, J items V, and L criteria M. We used conditional probability (CP) factorization (CP:CANDECOMP/PARAFAC), a tensor factorization method that decomposes the tensor into the sum of rank-one tensors:

$$R_{ij}^l = < U_i, V_j, M_l > = \sum_{k=1}^{K} u_{ki} \circ v_{kj} \circ m_{kl}. \tag{3}$$

$$\chi \simeq \sum_{k=1}^{K} u_k \circ v_k \circ m_k. \tag{4}$$

Each element x_{ijl} of χ can be calculated using $x_{ijl} \simeq \sum_{k=1}^{K} u_{ik} v_{jk} w_{lk}$, where $U = (u_{ik}) = (u_k)_{k=1}^{K} \in R^{I \times K}$, $V = (v_{jk}) = (v_k)_{k=1}^{K} \in R^{J \times K}$, and $M = (M_{lk}) = (m_k)_{k=1}^{K} \in R^{L \times K}$.

The latent representations of the users, items, and criteria are computed by minimizing the following regularized squared error from the observed ratings:

$$\min_{U,V,M} \sum_{(i,j,l) \in d} (R_{ij}^l - < U_i, V_j, M_l >)^2$$
$$+ \lambda(||U_i||^2 + ||V_j||^2 + ||M_l||^2), \tag{5}$$

where d is the set of observed user and item pairs of R, and constant λ works to avoid overfitting the observed evaluation ratings.

3. Bayesian probabilistic models

3.1. Bayesian probabilistic matrix factorization for collaborative filtering

Low-rank approximation methods are effective for CF and can generally perform efficiently on large datasets. However, a maximum a posteriori estimate (MAP) estimate of the model parameters needs to be found that conforms to the dataset because, if the regularization parameters are not tuned carefully, these models tend to overfit the data. Low-rank approximation methods can be generalized as a probabilistic model. The probabilistic matrix factorization (PMF) model introduces probabilistic distributions for matrix factorization (Mnih & Salakhutdinov, 2008).

A Bayesian PMF (BPMF) model (Salakhutdinov & Mnih, 2008) was proposed to provide probabilistic modeling in which model capacity is controlled by integrating over all parameters. Fig. 1 shows a schematic of the BPMF model. The conditional distribution $R = [R_{ij}]_{I \times J}$ for I users and J items and the latent features over $U \in R^{K \times I}$ and $V \in R^{K \times J}$ are given by

$$p(R|U, V, \alpha) = \prod_{i=1}^{I} \prod_{j=1}^{J} [\mathcal{N}(R_{ij}|U_i^T V_j, \alpha^{-1})]^{N_{ij}}. \tag{6}$$

$$p(U|\mu_U, \Lambda_U) = \prod_{i=1}^{I} \mathcal{N}(U_i|\mu_U, \Lambda_U^{-1}), \tag{7}$$

$$p(V|\mu_V, \Lambda_V) = \prod_{j=1}^{J} \mathcal{N}(V_j|\mu_V, \Lambda_V^{-1}), \tag{8}$$

where $\mathcal{N}(x|\mu, \alpha^{-1})$ denotes a Gaussian distribution with mean μ and precision α, and N_{ij} is a variable that is 1 if user i rated item j and 0 otherwise.

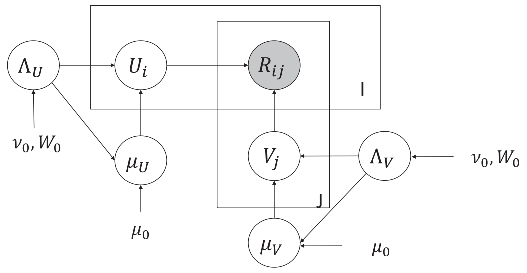

Fig. 1. Schematic of Bayesian probabilistic matrix factorization model.

The prior distributions from which the hyperparameters are obtained must be selected. For the Gaussian parameters, the conjugate distributions as priors are placed on the user and item hyperparameters: $\Theta_U = \{\mu_U, \Lambda_U\}$ and $\Theta_V = \{\mu_V, \Lambda_V\}$,

$$\begin{aligned} p(\Theta_U|\Theta_0) &= p(\mu_U|\Lambda_U)p(\Lambda_U) \\ &= \mathcal{N}(\mu_U|\mu_0, (\beta_0\Lambda_U)^{-1})\mathcal{W}(\Lambda_U|W_0, \nu_0), \end{aligned} \tag{9}$$

$$\begin{aligned} p(\Theta_V|\Theta_0) &= p(\mu_V|\Lambda_V)p(\Lambda_V) \\ &= \mathcal{N}(\mu_V|\mu_0, (\beta_0\Lambda_V)^{-1})\mathcal{W}(\Lambda_V|W_0, \nu_0). \end{aligned} \tag{10}$$

Here, $\mathcal{W}$ is a Wishart distribution with ν_0 and a $K \times K$ scale matrix W_0,

$$\mathcal{W}(\Lambda|W_0, \nu_0) = \frac{1}{G}|\Lambda|^{(\nu_0-K-1)/2}exp\Big(-\frac{1}{2}\mathrm{Tr}(W_0^{-1}\Lambda)\Big), \tag{11}$$

where G is a constant. Hyperpriors are defined by $\Theta_0 = (\mu_0, \nu_0, W_0)$.

The distribution of predicted ratings $\hat{R}_{ij}$ for user i and item j is computed by integrating over the model parameters:

$$\begin{aligned} p(\hat{R}_{ij}|R, \Theta_0) = \iint & p(\hat{R}_{ij}|U_i, V_j)p(U, V|R, \Theta_U, \Theta_V) \\ & p(\Theta_U, \Theta_V|\Theta_0)d\{U, V\}d\{\Theta_U, \Theta_V\}. \end{aligned} \tag{12}$$

Accurate estimation of this predictive distribution is difficult because it involves a multidimensional integral. We must thus rely on approximate inference. We use a Markov chain Monte Carlo (MCMC) method that is widely used for sampling. This method draws samples from a given distribution represented as a Markov chain. Then we approximate the integral with some samples in (12) using

$$p(\hat{R}_{ij}|R, \Theta_0) \approx \frac{1}{C}\sum_{c=1}^{C} p(\hat{R}_{ij}|U_i^{(c)}, V_j^{(c)}), \tag{13}$$

where C represents the number of samples acquired, and $\{U_i^{(c)}, V_j^{(c)}\}$ comes from the cth sample.

3.2. Bayesian probabilistic tensor factorization for temporal collaborative filtering

In the same way as described above for the BPMF model, a tensor factorization recommendation model can be generalized as a probabilistic model (Xiong et al., 2010). It was introduced the time latent features as a additional factors, and formulated as a tensor factorization based on the time dimension. This “Bayesian probabilistic tensor factorization (BPTF)” model works effectively for recommendation with several real-world datasets.

A rating can be denoted as R_{ij}^t, where index i, j denotes a user and item pair as above, and index t denotes the time slice in which the rating was given. With the BPMF model, the prior distributions of user latent features, item latent features, and time latent features are estimated to be Gaussian:

$$p(R|U, V, T, \alpha) = \prod_{i=1}^{I}\prod_{j=1}^{J}\prod_{t=1}^{D}[\mathcal{N}(R_{ij}^t| < U_i, V_j, T_t >, \alpha^{-1})]^{N_{ijt}} \tag{14}$$

$$T_1 = \mathcal{N}(\mu_T, \Lambda_T^{-1}). \tag{15}$$

$$T_l = \prod_{t=2}^{D}\mathcal{N}(T_{t-1}, \Lambda_T^{-1}). \tag{16}$$

The prior distributions of the user latent features and item latent features are the same as those given by (7) and (8). For the Gaussian parameters, the conjugate distributions as priors are placed on the user, item and time hyperparameters $\Theta_U = \{\mu_U, \Lambda_U\}$, $\Theta_V = \{\mu_V, \Lambda_V\}$, and $\Theta_T = \{\mu_T, \Lambda_T\}, \alpha$,

$$p(\alpha) = \mathcal{W}(\alpha|\hat{W}_0, \hat{\nu}_0), \tag{17}$$

$$\begin{aligned} p(\Theta_T|\Theta_0) &= p(\mu_T|\Lambda_T)p(\Lambda_T) \\ &= \mathcal{N}(\mu_T|\rho_0, (\beta_0\Lambda_T)^{-1})\mathcal{W}(\Lambda_T|W_0, \nu_0). \end{aligned} \tag{18}$$

The conjugate distributions of the users and items are the same as those given by (9) and (10). Here, precision α is defined as a tuning parameter. The distribution of predicted ratings $\hat{R}_{ij}^t$ for user i, item j, and time t is given by

$$\begin{aligned} p(\hat{R}_{ij}^t|R, \Theta_0) = \iint & p(\hat{R}_{ij}^t|U_i, V_j, T_t, \alpha)p(U, V, T, \alpha|R, \Theta_U, \Theta_V, \Theta_T) \\ & p(\Theta_U, \Theta_V, \Theta_T|\Theta_0)d\{U, V, T, \alpha\}d\{\Theta_U, \Theta_V, \Theta_T\}. \end{aligned} \tag{19}$$

The predictive distribution (19) cannot be computed analytically because it involves a multidimensional integral. Again, using an MCMC method to estimate the predictive distribution of (19):

$$p(\hat{R}_{ij}^t|R, \Theta_0) \approx \frac{1}{C}\sum_{c=1}^{C} p(\hat{R}_{ij}^t|U_i^{(c)}, V_j^{(c)}, T_t^{(c)}, \alpha^{(c)}), \tag{20}$$

where C denotes the number of samples collected, and $\{U_i^{(c)}, V_j^{(c)}, T_t^{(c)}, \alpha^{(c)}\}$ comes from the cth sample.

4. Bayesian probabilistic tensor factorization for multicriteria evaluation data

As indicated above, BPTF (Xiong et al., 2010) uses a special constraint on the time dimension. We have now introduced multicriteria latent features as additional factors and have formulated a tensor factorization model based on user, item, and multicriteria dimensions. A schematic of our Bayesian probabilistic tensor factorization for multicriteria (BPTF-MC) model is shown in Fig. 2. In this model, ratings are calculated using user latent features and item latent features. The multicriteria rating are modeled by calculating the ratings from the user latent features, item latent features, and multicriteria latent features. We denote a rating as R_{ij}^l, where i and j denote a user and an item as above, and l denotes the corresponding rating. The ratings are combined into a three-dimensional tensor, with the three dimensions corresponding to user, item, and criteria slices with sizes I, J, and L, respectively. This extension of the BPMF model leads to the assumption that the conditional distribution over the observed ratings $R = [R_{ij}^l]_{I\times J\times L}$

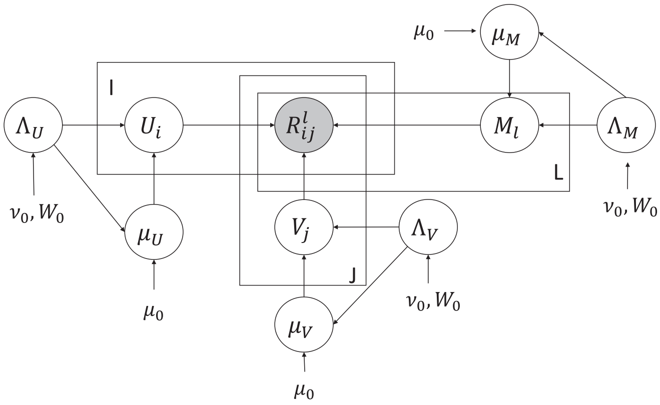

Fig. 2. Schematic of Bayesian probabilistic tensor factorization for multicriteria model.

and the latent features over $U \in R^{K \times I}$, $V \in R^{K \times J}$, and $M \in R^{K \times L}$ are Gaussian:

$$p(R|U, V, M, \alpha) = \prod_{i=1}^{I} \prod_{j=1}^{J} \prod_{l=1}^{L} [\mathcal{N}(R_{ij}^{l}| < U_i, V_j, M_l >, \alpha^{-1})]^{N_{ijl}} \tag{21}$$

$$p(M|\mu_M, \Lambda_M) = \prod_{l=1}^{L} \mathcal{N}(M_l|\mu_M, \Lambda_M^{-1}). \tag{22}$$

We use the Gaussian distribution and the Wishart distribution as prior distributions with references (Mnih & Salakhutdinov, 2008; Salakhutdinov & Mnih, 2008; Xiong et al., 2010). These distributions have been used in research on rating regression and have demonstrated good performance for rating prediction, making them well suited for our purposes. The prior distributions of user latent features and item latent features are the same as those given by (7) and (8). Again, the prior distributions from which the hyperparameters are obtained must be selected. For the Gaussian parameters, the conjugate distributions as priors are placed on the user, item, and criteria hyperparameters $\Theta_U = \{\mu_U, \Lambda_U\}$, $\Theta_V = \{\mu_V, \Lambda_V\}$, and $\Theta_M = \{\mu_M, \Lambda_M\}$:,

$$\begin{aligned} p(\Theta_M|\Theta_0) &= p(\mu_M|\Lambda_M)p(\Lambda_M) \\ &= \mathcal{N}(\mu_M|\mu_0, (\beta_0\Lambda_M)^{-1})\mathcal{W}(\Lambda_M|W_0, \nu_0). \end{aligned} \tag{23}$$

The conjugate distributions of users and items are the same as those given by (9) and (10), and the Wishart distribution is the same as that given by (11). As for the BPMF model, the distribution of predicted rating $\hat{R}_{ij}^{l}$ for user i, item j, and criteria l is computed by integrating over the model parameters:

$$\begin{aligned} p(\hat{R}_{ij}^{l}|R, \Theta_0) = \iint & p(\hat{R}_{ij}^{l}|U_i, V_j, M_l)p(U, V, M|R, \Theta_U, \Theta_V, \Theta_M) \\ & p(\Theta_U, \Theta_V, \Theta_M|\Theta_0)d\{U, V, M\}d\{\Theta_U, \Theta_V, \Theta_M\}. \end{aligned} \tag{24}$$

This predictive distribution cannot be computed analytically because it involves a multidimensional integral. We therefore again use an MCMC method. The predictive distribution of (24) is given by

$$p(\hat{R}_{ij}^{l}|R, \Theta_0) \approx \frac{1}{C} \sum_{c=1}^{C} p(\hat{R}_{ij}^{l}|U_i^{(c)}, V_j^{(c)}, M_l^{(c)}), \tag{25}$$

where C denotes the number of samples collected, and $\{U_i^{(c)}, V_j^{(c)}, M_l^{(c)}\}$ comes from the cth sample.

There are quite a few MCMC methods. We used the Gibbs sampling algorithm, which cycles through the latent variables, sampling each one from its distribution conditional on the current values of all other variables. Gibbs sampling is suitable for such conditional distributions. We first consider the user features. The conditional distribution over the user feature vectors and the user hyperparameters is Gaussian:

$$p(U \mid V, M, R, \Theta_U) = \prod_{i=1}^{I} p(U_i \mid V, M, R, \Theta_U), \tag{26}$$

$$p(U_i \mid V, M, R, \Theta_U) = \mathcal{N}(U_i \mid \mu_i^*, (\Lambda_i^*)^{-1}), \tag{27}$$

$$\mu_i^* \equiv (\Lambda_i^*)^{-1}(\Lambda_U\mu_U + \alpha \sum_{j=1}^{J} \sum_{l=1}^{L} I_{ij}^{l} R_{ij}^{l} Q_{jl}), \tag{28}$$

$$\Lambda_i^* \equiv \Lambda_U + \alpha \sum_{j=1}^{J} \sum_{l=1}^{L} I_{ij}^{l} Q_{jl} Q_{jl}^{'}, \tag{29}$$

$$p(\mu_U, \Lambda_U \mid U) = \mathcal{N}(\mu_U \mid \mu^*, (\beta_0^*\Lambda_U)^{-1})\mathcal{W}(\Lambda_U \mid W_0^*, \nu_0^*), \tag{30}$$

where

$$\begin{aligned} \mu_0^* &= \frac{\beta_0\mu_0 + I\hat{U}}{\beta_0 + I}, \beta_0^* = \beta_0 + I, \nu_0^* = \nu_0 + I, \\ (W_0^*)^{-1} &= W_0^{-1} + I\hat{S} + \frac{\beta_0 I}{\beta_0 + I}(\mu_0 - \hat{U})(\mu_0 - \hat{U})^T, \\ \hat{U} &= \frac{1}{I}\sum_{i=1}^{I} U_i, \hat{S} = \frac{1}{I}\sum_{i=1}^{I} U_i U_i^T. \end{aligned}$$

$Q_{jl} \equiv V_j \cdot M_l$ is the element-wise product of V_j and M_l. The conditional distributions over the item (criteria) feature vectors and the item (criteria) hyperparameters have exactly the same form.

$$p(V_j \mid U, M, R, \Theta_V) = \mathcal{N}(V_j \mid \mu_j^*, (\Lambda_j^*)^{-1}), \tag{31}$$

$$\mu_j^* \equiv (\Lambda_j^*)^{-1}\left(\Lambda_V\mu_V + \alpha \sum_{i=1}^{I} \sum_{l=1}^{L} I_{ij}^{l} R_{il}^{l} P_{il}\right), \tag{32}$$

$$\Lambda_j^* \equiv \Lambda_V + \alpha \sum_{i=1}^{I} \sum_{l=1}^{L} I_{ij}^{l} P_{il} P_{il}^{'}, \tag{33}$$

$$p(M_l \mid U, V, R, \Theta_M) = \mathcal{N}(M_l \mid \mu_l^*, (\Lambda_l^*)^{-1}), \tag{34}$$

$$\mu_l^* \equiv (\Lambda_l^*)^{-1}\left(\Lambda_M\mu_M + \alpha \sum_{i=1}^{I} \sum_{j=1}^{J} I_{ij}^{l} R_{ij}^{l} H_{ij}\right), \tag{35}$$

$$\Lambda_t^* \equiv \Lambda_M + \alpha \sum_{i=1}^{I} \sum_{j=1}^{J} I_{ij}^{l} H_{ij} H_{ij}^{'}, \tag{36}$$

where $P_{il} \equiv U_i \cdot M_l$ is the element-wise product of U_i and M_l, and $H_{ij} \equiv U_i \cdot V_j$ is the element-wise product of U_i and V_j.

The Gibbs sampling algorithm is presented in Algorithm 1.

Algorithm 1 Gibbs sampling for BPTF-MC.

```
Initialize model parameters $\{U^{(1)}, V^{(1)}, M^{(1)}\}$.
for $c = 1$ to $C$ do
    • Sample the hyperparameters (9),(10),(23):
    $\Theta_U^{(c)} \sim p(\Theta_U^{(c)} \mid U^{(c)})$,
    $\Theta_V^{(c)} \sim p(\Theta_V^{(c)} \mid V^{(c)})$,
    $\Theta_M^{(c)} \sim p(\Theta_M^{(c)} \mid M^{(c)})$.
    for $i = 1$ to $I$ do
        • Sample user features in parallel (27):
        $U_i^{(c+1)} \sim p(U_i \mid V^{(c)}, M^{(c)}, \Theta_U^{(c)}, R)$.
    end for
    for $j = 1$ to $J$ do
        • Sample item features in parallel (31):
        $V_j^{(c+1)} \sim p(V_j \mid U^{(c+1)}, M^{(c)}, \Theta_V^{(c)}, R)$.
    end for
    for $l = 1$ to $L$ do
        • Sample criteria features in parallel (34):
        $M_l^{(c+1)} \sim p(M_l \mid U^{(c+1)}, V^{(c+1)}, \Theta_M^{(c)}, R)$.
    end for
end for
```

5. Experiment

5.1. Recommendation

5.1.1. Experimental settings

We investigated the performance of five collaborative filtering models (three single-criterion models and two multicriteria models), including our extended Bayesian probabilistic tensor factorization for multicriteria model. We used the Rakuten Travel dataset

Table 1
Datasets Used for Recommendation Experiment.

Dataset	No. of Users	No. of Items	No. of Ratings
Rakuten Travel (Hotels)	881	5,098	16,993
Rakuten GORA (Golf courses)	8,366	1,220	62,115

and the Rakuten GORA dataset for large and sparse multicriteria evaluation data, which are available online[1]. The Rakuten Travel dataset includes hotel data and review comments. Each reviewer provided an overall rating for the hotel along with ratings for six criteria (*location, room, meals, bath, service*, and *equipment*). The Rakuten GORA dataset includes golf course data and review comments. Each reviewer provided an overall rating for the golf course along with ratings for seven criteria (*customer relations, course, meals, distance, cost-performance, fairways*, and *equipment*). The details of for the each datasets are shown in Table 1. The total number of ratings is the number of ratings times the number of ratings from different viewpoints. The ratings are on a scale of 1 to 5, with 5 being the best. We randomly extracted 10% of the items in each dataset and assumed that they had not been evaluated. These "unevaluated" items were used as test data, and the overall ratings were predicted. The inputs for the ratings were the integers 1 to 5, with 5 being the best, and the output predicted ratings were real numbers from 0 to 5. The evaluation metric was the root mean square error (RMSE), which is widely used for rating prediction.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{(i,j)}(R_{ij} - \hat{R}_{ij})^2}, \tag{37}$$

where N is the number of ratings in the test data, R_{ij} is an observed rating, and $\hat{R}_{ij}$ is a predicted rating. The five methods evaluated were as follows.

Matrix Factorization This method characterizes both users and items by using latent factors from ratings. Ratings are computed from the inner product of user latent factors and item latent factors.

Matrix Factorization (Biased) This method adds biases to matrix factorization.

Bayesian Probabilistic Matrix Factorization (BPMF) This method provides probabilistic modeling in which model capacity is controlled by integrating over all parameters.

Tensor Factorization This method is similar to matrix factorization except that it characterizes items, users, and multicriteria by using latent factors from ratings, which are computed from user latent factors, item latent factors, and multicriteria latent factors.

Bayesian Probabilistic Tensor Factorization for Multicriteria (BPTF-MC) The proposed method, which uses probabilistic modeling in which model capacity is controlled by integrating over all parameters.

For Matrix Factorization, Matrix Factorization (Biased), and BPMF, only the overall user ratings for items were input, and only the overall user ratings were predicted as output. The matrix formed by the overall ratings was decomposed into latent representations of users and items. The overall user rating for an item was computed as the product of the latent vectors of the user and item. For Tensor Factorization and BPTF-MC, the overall ratings and ratings from different user viewpoints for items were input, and these ratings were predicted simultaneously as output. These ratings were combined into a three-dimensional tensor with the three dimensions corresponding to user, item, and multicriteria. The user rating for an item by criteria was computed as the product of the user, item, and criteria latent vectors. To enable comparison of the single-criterion and multicriteria methods, only the overall user ratings were used as test data. We compared the accuracy of the ratings predicted by the methods that consider only the overall ratings with that of the ratings predicted by the methods that consider all the ratings.

5.1.2. Results and discussion

Rakuten Travel dataset. We compared the performance of the probabilistic methods with those of the low-rank approximation methods, focusing on Matrix Factorization, BPMF, Tensor Factorization, and BPTF-MC. The parameters used for the priors were fixed at $\alpha = 2$, $\mu_0 = 0$, $\nu_0 = K$, and W_0 as the identity matrix for both user and item and for multicriteria hyperpriors. As shown in Fig. 3, for Matrix Factorization and BPMF and for Tensor Factorization and BPTF-MC, the probabilistic methods performed much better than the low-rank approximation methods, which obtain a MAP estimate by fitting to the data. The proposed method, which uses multicriteria evaluation data, performed better than the matrix factorization methods, which use single-criterion evaluation.

Rakuten GORA dataset. We compared the performance of the probabilistic methods with that of the low-rank approximation methods, focusing on Matrix Factorization, BPMF, Tensor Factorization, and BPTF-MC. We used the same parameters as for the Travel dataset experiment. As shown in Fig. 4, for Matrix Factorization and BPMF and for Tensor Factorization and BPTF-MC, the probabilistic methods performed much better than the low-rank approximation methods, which, as mentioned above, obtain a MAP estimate by fitting to the data. Again, we observed that the proposed method, which, as mentioned above, uses multicriteria evaluation data, performed better than the matrix factorization methods, which use single-criterion evaluation.

5.2. Rating aggregation

The simplest method for aggregating the user ratings of an item is to average the ratings. However, if the number of users is very small, the aggregated rating is affected by the ratings of specific users. Moreover, if the reliabilities of the ratings are low, the reliability of the aggregated rating is also low. For this reason, many Web sites do not display an aggregated rating if the number of evaluators is small.

Several studies on reliably aggregating ratings from people in general (”worker” in the parlance of crowdsourcing) have focused on binary or multi-class labeling. They include ones that considered worker ability (Dawid & Skene, 1979), problem difficulty (Whitehill, Wu, Bergsma, Movellan, & Ruvolo, 2009), and worker confidence (Oyama, Baba, Sakurai, & Kashima, 2013). Other studies have focused on multilabeling of data wherein a data item can have multiple labels at the same time (Duan, Oyama, Sato, & Kurihara, 2014). Several studies have focused on the results of aggregating rating data (Uebersax & Grove, 1993). The conventional approaches to rating aggregation are based on the premise that a single rating is acceptable and use a probabilistic model. In our experiment of the efficiency of CF of multicriteria data, we regarded rating aggregation as an information recommendation problem and considered the "average user."

5.2.1. Experimental setup

To evaluate the performance of rating aggregation, we used various CF methods for rating aggregation. Since the Rakuten datasets do not contain aggregated ratings, we extracted the data for users who had evaluated many items and the data for items that had

[1] https://rit.rakuten.co.jp/data_release/.

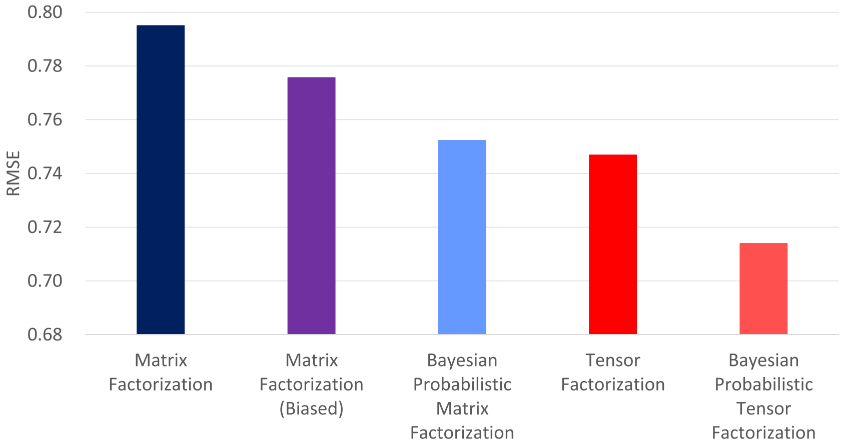

Fig. 3. RMSE for hotel recommendation.

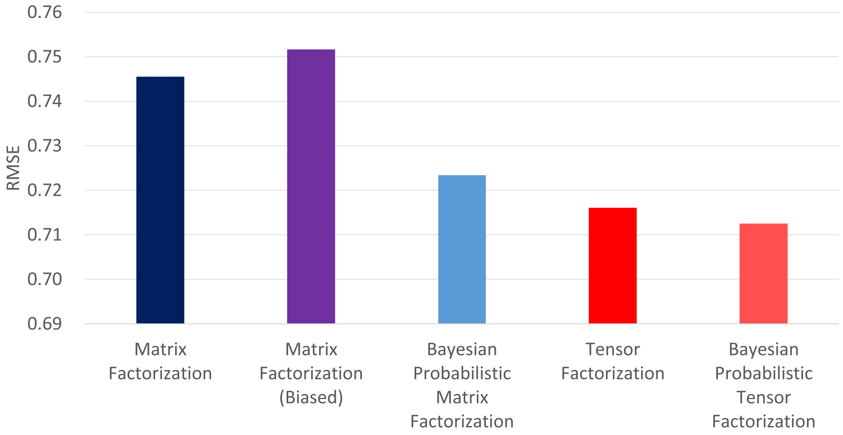

Fig. 4. RMSE for golf course recommendation.

been evaluated by many users. The aggregated rating for each extracted item was taken as the average of the ratings by the users who had evaluated the item. Because these items had been evaluated by many users, we assumed that the aggregated ratings were trustworthy and thus could be used as a gold standard. Then we added to the dataset an *average user* whose ratings were the aggregated ratings. We regarded rating aggregation as an item recommendation problem for the *average user* and evaluated prediction accuracy by using CF between randomly selected known users and the *average user*. We calculated the aggregated ratings from the evaluation of a small number of users. To measure prediction accuracy, we again used the RMSE. As we did in the CF experiment, for Matrix Factorization, Matrix Factorization (Biased), and BPMF, only the overall user ratings for items were input, and only the overall ratings of the *average user* were predicted as output. For Tensor Factorization and BPTF-MC, the overall ratings and ratings from different viewpoints were input, and these ratings of the *average user* were predicted simultaneously as output. Again, to compare the methods that consider only the overall ratings with the ones that consider multicriteria ratings, only the overall ratings of the *average user* were used as test data. We compared the accuracy of the aggregated ratings predicted by the methods that consider only the overall ratings with that of the aggregated ratings predicted by the methods that consider multicriteria ratings.

5.2.2. Results and discussion

Figs. 5 and 6 show the results for two, three, four, and five known users, i.e., the users who evaluated the items for which the aggregated rating is to be predicted. The smaller the number of known users, the smaller the amount of information about the item, which makes it more difficult to predict the aggregated rating.

Rakuten Travel dataset. To create aggregated ratings from the Rakuten Travel dataset, we extracted the data for users who had evaluated 15 or more hotels and for hotels that had been evaluated by 15 or more users. We thereby obtained 76 aggregated hotel ratings; 80% were used for training data and the remaining 20% were used for test data.

Then, as we did in the CF experiment, we compared the probabilistic methods to the low-rank approximation methods using the same parameters. As shown in Fig. 5, BPMF had better performance than Matrix Factorization. Moreover, BPTF-MC had better performance than Tensor Factorization, as in the CF experiment. Furthermore, BPTF-MC had better (or similar) performance than BPMF. It is thus also effective to consider multicriteria evaluation rather than single-criterion evaluation for probabilistic methods.

Rakuten GORA dataset. To create aggregated ratings from the Rakuten GORA dataset, we extracted the data for users who had evaluated 30 or more golf courses and for golf courses that had been evaluated by 30 or more users. We thereby obtained 519 aggregated golf course ratings; 80% were used for training data and the remaining 20% were used for test data.

Then, as in the Travel dataset experiment, we compared the probabilistic methods to the low-rank approximation methods using the same parameters. As shown in Fig. 6, BPMF had better

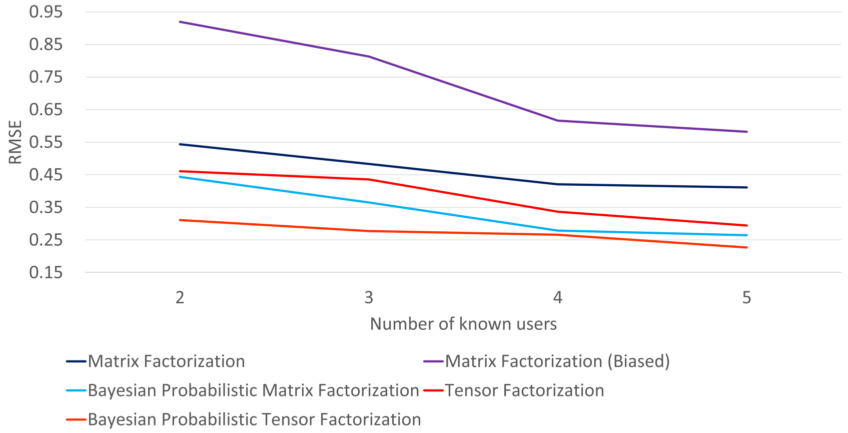

Fig. 5. RMSE for hotel rating aggregation.

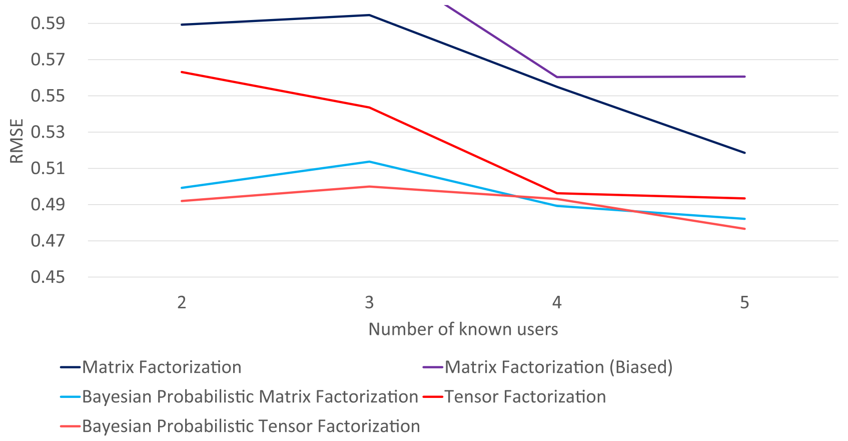

Fig. 6. RMSE for golf course rating aggregation.

performance than Matrix Factorization. Moreover, BPTF-MC had better performance than Tensor Factorization, as in the CF experiment. Furthermore, BPTF-MC had better (or similar) performance than BPMF. So again, it is more effective to consider multicriteria rating than single-criterion rating for probabilistic methods.

6. Conclusion

We formulated a Bayesian probabilistic tensor factorization for multicriteria (BPTF-MC) model that uses multicriteria evaluation by placing hyperpriors over the hyperparameters and using a Markov chain Monte Carlo method to perform approximate inference. BPTF-MC can process more detailed information than matrix factorization due to the addition of a set of multicriteria evaluations and the use of Bayesian inference rather than parameter tuning. BPTF-MC performed better than low-rank approximation methods, which obtain a MAP estimate by fitting to the data, for recommendation for rating aggregation using large and sparse multicriteria evaluation data. Our evaluation showed that considering multicriteria rating is more effective than considering single-criterion rating for probabilistic methods because BPTF-MC can predict ratings more accurately than BPMF both for recommendation and rating aggregation.

Future work includes using latent Dirichlet allocation (Blei, Ng, & Jordan, 2003) to analyze review comments. It also includes investigating the use of other approaches such as collaborative topic modeling (Wang & Blei, 2011) and neural collaborative filtering (He et al., 2017), which apply recommendation techniques to deep neural networks.

Credit authorship contribution statement

Hiroki Morise: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Satoshi Oyama:** Conceptualization, Methodology, Validation, Formal analysis, Resources, Writing - review & editing, Supervision, Project administration, Funding acquisition. **Masahito Kurihara:** Conceptualization, Validation, Formal analysis, Resources, Writing - review & editing, Supervision, Project administration.

Acknowledgments

The Rakuten Travel and GORA datasets were provided by Rakuten, Inc. through the National Institute of Informatics. This work was partially supported by a Japan Society for the Promotion of Science KAKENHI Grant (JP15H02782).

References

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Dawid, A. P., & Skene, A. M. (1979). Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society. Series C, 28*(1), 20–28. doi:10.2307/2346806.

Duan, L., Oyama, S., Sato, H., & Kurihara, M. (2014). Separate or joint? estimation of multiple labels from crowdsourced annotations. *Expert Systems with Applications, 41*(13), 5723–5732. doi:10.1016/j.eswa.2014.03.048.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T.-S. (2017). Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web* (pp. 173–182).

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer, 8*(8), 30–37.

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

Morise, H., Oyama, S., & Kurihara, M. (2017). Collaborative filtering and rating aggregation based on multicriteria rating. In *Proceedings of the first ieee workshop on human-machine collaboration in big data (hmdata)* (pp. 4335–4340). doi:10.1109/bigdata.2017.8258477.

Oyama, S., Baba, Y., Sakurai, Y., & Kashima, H. (2013). Accurate integration of crowdsourced labels using workers' self-reported confidence scores. In *International joint conference on artificial intelligence* (pp. 2554–2560).

Salakhutdinov, R., & Mnih, A. (2008). Bayesian probabilistic matrix factorization using Markov Chain Monte Carlo. In *Proceedings of the 25th international conference on machine learning* (pp. 880–887). doi:10.1145/1390156.1390267.

Uebersax, J. S., & Grove, W. M. (1993). A latent trait finite mixture model for the analysis of rating agreement. *Biometrics, 49*(3), 823–835. doi:10.2307/2532202.

Wang, C., & Blei, D. M. (2011). 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). doi:10.1145/2020408.2020480.

Whitehill, J., Wu, T.-f., Bergsma, J., Movellan, J. R., & Ruvolo, P. L. (2009). Whose vote should count more: optimal integration of labels from labelers of unknown expertise. In *Neural information processing systems* (pp. 2035–2043).

Xiong, L., Chen, X., Huang, T.-K., Schneider, J., & Carbonell, J. G. (2010). Temporal collaborative filtering with Bayesian probabilistic tensor factorization. In *Proceedings of the 2010 siam international conference on data mining* (pp. 211–222). doi:10.1137/1.9781611972801.19.