Computer Science and Information Systems 18(1):285–301 https://doi.org/10.2298/CSIS200531056H

Hypothetical Tensor-based Multi-criteria Recommender System for New Users with Partial Preferences

Minsung Hong[1] and Jason J. Jung[2]*

[1] Western Norway Research Institute
Box 163, NO-6851 Sogndal, Norway
msh@vestforsk.no

[2] Department of Computer Engineering
Chung-Ang University
84 Heukseok-ro, Seoul, Korea
j3ung@cau.ac.kr, j2jung@gmail.com

Abstract. Multi-Criteria Recommender Systems (MCRSs) have been developed to improve the accuracy of single-criterion rating-based recommender systems that could not express and reflect users' fine-grained rating behaviors. In most MCRSs, new users are asked to express their preferences on multi-criteria of items, to address the cold-start problem. However, some of the users' preferences collected are usually not complete due to users' cognitive limitation and/or unfamiliarity on item domains, which is called 'partial preferences'. The fundamental challenge and then negatively affects to accurately recommend items according to users' preferences through MCRSs. In this paper, we propose a Hypothetical Tensor Model (HTM) to leverage auxiliary data complemented through three intuitive rules dealing with user's unfamiliarity. First, we find four patterns of partial preferences that are caused by users' unfamiliarity. And then the rules are defined by considering relationships between multi-criteria. Lastly, complemented preferences are modeled by a tensor to maintain an inherent structure of and correlations between the multi-criteria. Experiments on a TripAdvisor dataset showed that HTM improves MSE performances from 40 to 47% by comparing with other baseline methods. In particular, effectivenesses of each rule regarding multi-criteria on HTM are clearly revealed.

Keywords: Cold-start problem, Partial preferences, Multi-criteria recommender system, Tensor factorization.

1. Introduction

The amount of valuable data available on the Internet and the number of its users have hugely increased in the last decades. Although the data can be helpful to the users who try to find useful information such as restaurants, hotels, and museums appropriated to their interests, results provided by a search engine may be overwhelming on the Web or relevant applications [3,19]. Therefore, recommender systems have been broadly studied to cope with the information overload by providing personalized recommendations, content, and services. For instance, such systems automatically extract tourists' preferences from their explicit or implicit feedback and match features of tourism items with their needs [7,29,31].

* Corresponding author

Conventional recommendation techniques such as Collaborative Filtering (CF), as one of the most well-known and frequently adopted methods to recommend items in various fields, are typically developed based on a single rating type. However, such single-criterion recommender systems can not express and reflect fine-grained user rating behaviors, and the accuracy of those systems is often low [2]. For example, in some cases (e.g., restaurant or hotel recommendations), multiple ratings (e.g., overall, staff, or atmosphere) can often be collected to reflect various aspects of restaurants and hotels. Such multiple-criteria data would be a source of rich intelligence on item recommendations, if the data is appropriately analyzed and applied.

However, it is a non-trivial task to exploit the multiple ratings into recommendation services due to the cold-start problem that becomes more severe in the context of MCRSs. In most of the systems, new users are asked to present their preferences on some criteria of items in order to address the cold-start problem, and it could be lots of burden to users. Furthermore, some of the preferences collected are incompletely answered due to the users' cognitive limitation and/or unfamiliarity on item domains, which are called 'partial preferences' [23]. The fundamental problem thus results in low performances of MCRSs.

In this paper, we find four patterns of the 'partial preference' via data analysis in the context of MCRSs. And then a Hypothetical Tensor Model (HTM) based on three rules managing unknown users' preferences in the four patterns is proposed and is used to predict users' unobserved ratings through the Higher-Order Singular Value Decomposition (HOSVD). Simultaneously, the model keeps inherent correlations between multiple criteria. It is important using a tensor to model multiple user preferences and apply the defined rules into the model since the intuitive rules are introduced by considering relationships between users' unknown and known rating scores (i.e., multiple criteria). Experiments with a real-world dataset from Tripadvisor, which is one of the famous web review services for restaurant and hotel in tourism, show that HTM significantly outperforms than other baseline methods. Furthermore, we reveal the effectivenesses of three rules for each criterion rating. Therefore, our contributions of this paper are as follows.

- We find four patterns of partial preferences that are caused by new users' unfamiliarity in MCRSs.
- An intuitive rule set based on relationships between multi-criteria is defined to address the negative impact of unknown user preferences in the recognized patterns.
- We propose a rule-based hypothetical tensor model to improve the performance of MCRSs along with to maintain a structure of and correlations between multi-criteria
- Experimental results show better performances of the proposed method than baseline methods as well as effectivenesses of the proposed rules for each criterion.

The rest of this paper is organized as follows: In Section 2, we reviews relevant works associated with MCRSs. Section 3 introduces three intuitive rules to manage four patterns of partial preferences and proposes the hypothetical tensor model. In Section 4, we present experimental setup and evaluation protocol, while Section 5 describes in detail our empirical studies and discusses experimental results. Finally, Section 6 concludes with directions of future work.

2. Related work

Typical CF methods exploit a single rating as elements of a item-user matrix. Such techniques focus on one type (i.e., overall) of rating provided by users and suggest items to them based on preferences of their neighbors who have similar rating behaviors. Although such single rating-based approaches show a smooth and satisfying performance, as the appearance of multi-criteria recommendation techniques, one has been perceived that single criteria systems have relatively less accurate [18,5,28]. Thus, many researchers have studied to propose a new model for MCRSs [4,30,17,27].

The multi-criteria approach can be classified into memory-based and model-based methods, like CF techniques. In memory-based approach, similarities are mainly computed in two ways: one combines traditional similarity values for each criterion into a single similarity through aggregation methods (e.g., average and weighted sum) [1]. The other approach calculates distances between multi-criteria directly via multi-dimensional distance metrics (e.g., Euclidean and Manhattan). The model-based approach builds a model to predict unknown ratings and is based on the assumption that an item rating doesn't independent with other ratings and there exist relations between multi-criteria ratings. In this regard, various techniques have been used such as probabilistic modeling [25], support vector regression, multi-linear singular value decomposition [9], and genetic algorithm [10], deep neural network [22].

However, although an expected improvement of MCRSs could be achieved under the idea that such systems can obtain abundantly ratings for multi-criteria [21,2], it is generally difficult to get complete preferences due to users' unfamiliarity [23], such as four patterns we found in this paper, for rating scheme in real-world systems. Furthermore, it is a non-trivial task to exploit multiple ratings because of correlations between them. Therefore, MCRSs need to maintain a structure of multi-criteria and correlations between them when such systems model user preferences.

Therefore, in this paper, we find patterns of unknown rating class that are caused by new users' unfamiliarity on a rating scheme and define intuitive rules to complement the incomplete data by considering relationships between multi-criteria. And then the user preferences are expressed by a tensor model to keep the inherent structure of and the relationship between multiple ratings. Note that the proposed model is able to be applied into various domains having multiple criteria, such as restaurant, hotel and POI recommendation services.

3. Partial preference and hypothetical tensor factorization

3.1. Complementation with unknown class rules

As above-mentioned, in the context of MCRSs, users are often asked to fill multiple criteria for evaluating items based on their experiences. However, such tasks are burdens for users and result in incomplete multiple ratings due to users' unfamiliarity on the evaluation scheme. To alleviate the above problem, we intuitively define three rules for unknown rating class by an analysis of users' preferences on multiple criteria with examples. Figure 1 illustrates the reason how MCRSs can be more accurate than single-criterion based recommendations and presents potential patterns caused by users who are unfamiliar to

	Item i_1	Item i_2	Item i_3	Item i_4	Item i_5
User u_1	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	?
User u_2	$3_{(2,2,4,4)}$	$3_{(1,1,5,5)}$	$3_{(2,2,4,4)}$	$3_{(1,1,5,5)}$	$2_{(1,1,3,2)}$
User u_3	$4_{(5,5,3,3)}$	$2_{(3,3,1,1)}$	$4_{(4,4,3,3)}$	$2_{(3,3,1,1)}$	$4_{(5,5,3,3)}$

(a) Single criterion vs multiple criteria

	Item i_1	Item i_2	Item i_3	Item i_4	Item i_5
User u_1	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	?
User u_4	$5_{(5,5,5,5)}$	$1_{(1,1,1,1)}$	$5_{(5,5,5,5)}$	$1_{(1,1,1,1)}$	$2_{(1,1,3,2)}$
User u_5	$4_{(5,5,3,3)}$	$2_{(3,3,1,1)}$	$4_{(5,5,3,3)}$	$2_{(3,3,1,1)}$	$4_{(5,5,5,5)}$

(b) Only or without overall criterion

	Item i_1	Item i_2	Item i_3	Item i_4	Item i_5
User u_1	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	?
User u_6	$3_{(2,2,4,4)}$	$2_{(0,1,2,3)}$	$3_{(2,2,4,4)}$	$2_{(1,0,2,3)}$	$2_{(0,0,2,2)}$
User u_7	$3_{(2,2,4,4)}$	$2_{(1,2,3,0)}$	$3_{(2,2,4,4)}$	$2_{(1,2,3,0)}$	$4_{(5,5,3,3)}$

(c) Zero used to the worst criterion

	Item i_1	Item i_2	Item i_3	Item i_4	Item i_5
User u_1	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	$4_{(3,3,5,5)}$	$2_{(1,1,3,3)}$	?
User u_8	$4_{(0,3,4,5)}$	$2_{(0,1,2,3)}$	$4_{(0,3,4,5)}$	$2_{(0,1,2,3)}$	$2_{(0,3,2,1)}$
User u_9	$3_{(2,3,4,0)}$	$2_{(1,2,3,0)}$	$3_{(2,3,4,0)}$	$2_{(1,2,3,0)}$	$4_{(5,5,3,0)}$

(d) Consideration partially

Fig. 1. Partial preference patterns

the rating scheme of MCRSs. Given a rating scheme ranged from 1 (worst) to 5 (best), let's assume that five ratings including overall rating are obtained for multi-criteria. In the (a) of the figure, a recommender system using only overall rating will select the user u_3, who has similar rating behaviors to the target user u_1. However, the user u_3 oppositely rates multi-criteria of the items by comparing with the target user actually. On the other hand, user u_2 has more similar behaviors with of user u_1, than the user u_3 in the context of MCRSs. Thus, the accuracy of MCRSs is often higher than of single-criterion based recommendations. Although this improvement can be achieved under the assumption that such systems can obtain abundantly ratings for multi-criteria, it is difficult to get complete preferences due to users' unfamiliarity for rating scheme in fact. Even though, these unknown ratings will be predicted by MCRSs, it makes sparsity problem severe.

As shown in the Figure, we found four patterns, which frequently caused by the unfamiliarity issue, via data analysis of users' multiple ratings. The patterns illustrated by the examples (b), (c), and (d) in Figure 1 are as follows. In the examples, ratings with gray color in brackets indicate unknown values.

- The first and second patterns are that users often input only or without overall ratings. There are two sub-cases of the input case of only overall ratings: unambiguous and ambiguous ratings. The former includes ratings 1 and 5 considered as clearly worst and best evaluations. The other contains other ratings (i.e., 2, 3, and 4 in the above-mentioned rating scheme). For instance, it could be clearly considered that the unknown ratings of user u_4 are 5 or 1. In this regard, the user u_4 will be used to predict unknown ratings of user u_1.
- The other cases are less occurred than the above one, but it affects the accuracy of MCRSs also. The third one is that users sometimes do not input any ratings to express their worst preference. This pattern has mainly occurred with low overall ratings. In the case of example (c), let's assume that the users u_6 and u_7 used zero (not input any ratings) to show their worst experience. As a result, although u_7 has different multiple ratings with the target user u_1, both users selected on the recommendation process if a MCRS does not take this pattern into account.

– The last one relates to users who do not consider some of multi-criteria. It could happen due to individual users' criteria or ambiguous experience. That is, a user may do not enter some ratings because she/he does not care the relevant criteria or needs too much effort because of difficult decisions. In this regard, if users u_8 and u_9 do not care the first and last criterion respectively, selecting only the user u_8 would be appropriate to predict preferences of the target user u_1 to item i_5.

To alleviate negative effects of the above patterns, we propose three rules that are based on other filled multiple ratings (i.e., relationships between multi-criteria). Given overall rating r_0 and n number of the other multiple ratings $r_{k \in \{1,2,\ldots,n\}}$ with a rating scheme (1 to N), a generalized rule function $\mathcal{RU}$ is defined by

$$\mathcal{RU}() = \begin{cases} r_0 & \mathcal{R}_1\text{: if } \sum_{k=1}^{n}[r_k{=}\emptyset]{=}n \wedge (r_0{=}1 \vee r_0{=}N) \\ 1 & \mathcal{R}_2\text{: if } \sum_{k=0}^{n}[r_k{=}\emptyset] \leq (n{+}1)/2 \wedge \arg(r_{\forall k}) \leq N \times 0.25 \\ N & \mathcal{R}_3\text{: if } \sum_{k=0}^{n}[r_k{=}\emptyset] \leq (n{+}1)/2 \wedge \arg(r_{\forall k}) \geq N \times 0.75, \end{cases} \tag{1}$$

where the $[\mathcal{P}]$ indicates the "Inverson bracket notation" that returns 1 if the condition $\mathcal{P}$ is true. As a summary, the $\mathcal{R}_1$ is mainly applied to the first pattern. While the second and last patterns are handled by the $\mathcal{R}_2$ and $\mathcal{R}_3$, the third one is managed by $\mathcal{R}_2$.

3.2. Hypothetical tensor model

This section describes a structure of the proposed HTM based on the rule function above-defined. Traditional Matrix Factorization (MF) techniques based on a two-dimensional user-item matrix are based on the idea that the overall ratings are generated by users' and items' latent factors. However, the assumption may fail to comprehensively represent a structure of and relationships between the latent factors [6], since it neglects considering multiple factors. Whereas, tensor models as a matrix generalization have been used to predict missing ratings along with maintaining a multi-dimensional structure of data, as it can consider the interdependency between multiple factors such as users, items, contexts, and so on in the research field of recommender systems [24,13,12,11].

In this paper, the context factors indicate multiple-criteria (i.e., rating types). Therefore, the HTM simply has three orders (i.e., user $\times$ item $\times$ rating type) as shown in (a) of Figure 2. The illustrations (b) and (c) represent a normal model and a HTM based on the proposed rule for unknown ratings in four patterns of partial preferences. In these examples, users and items are equal to of patterns in Figure 1 and the number of multiple ratings except for overall rating is 4. Note that we only illustrate users relate to the proposed rules to save space. White color represents unknown ratings. Also, light and dark gray colors indicate ratings filled by users and ratings complemented by the proposed rules, respectively. As a result, the density of the proposed hypothetical tensor model becomes higher than that of the normal tensor, and it helps to improve the accuracy of MCRSs as will be showed in Section 5.

3.3. Tensor factorization

This section defines factorization problem of the proposed HTM on prediction of unobserved users' preferences to items. Given I users, J items, and K rating types (i.e.,

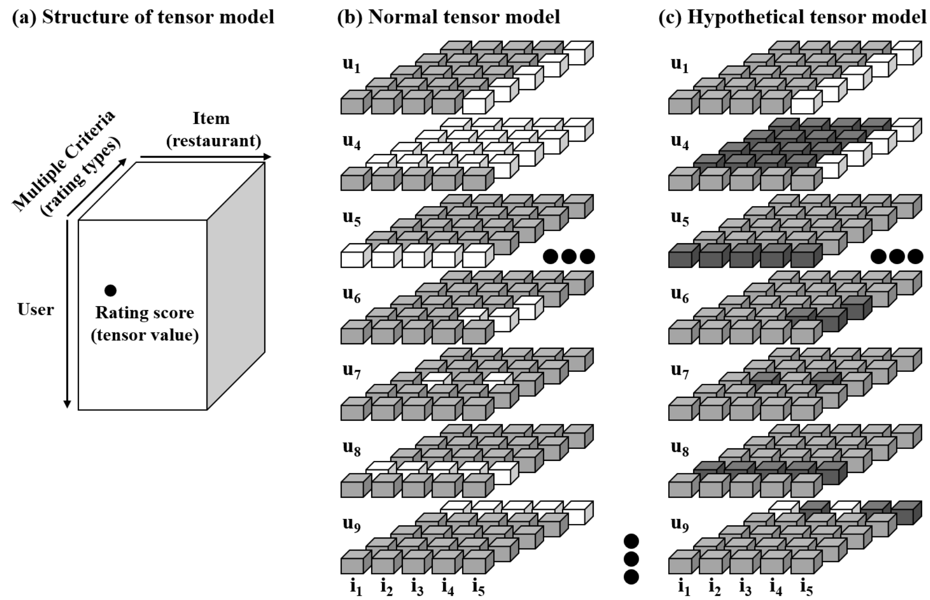

Fig. 2. Structure of rule-based hypothetical tensor model

multi-criteria including overall criterion), the proposed model $\mathcal{H}$ is defined as follows:

$$\mathcal{H} = \{H_{ijk}\} \in \mathbb{R}^{I \times J \times K}, \tag{2}$$

where the value H_{ijk} indicates a k^{th} rating of i^{th} user for j^{th} item.

Like a conventional tensor factorization, our problem is how to minimize loss between observed and approximate tensors with considering regularization risks. Therefore, we aim to minimize a loss function $L(H_{ijk}, \hat{H}_{ijk})$, where the original and approximate users' ratings for k^{th} criterion of items are H_{ijk} and $\hat{H}_{ijk}$. For better generalization performances, a regularization term $\Omega(H_{ijk})$ is also added to the loss function. Thus, a final loss function is $L(H_{ijk}, \hat{H}_{ijk}) + \Omega(H_{ijk})$ along with least squares loss function $L()$ and Frobenius norm Ω as standard choices. Also, $\times_U$ represents a tensor-matrix multiplication operator, where the subscript indicates a direction of the tensor on which the matrix is multiplied. Additionally, entries of a i^{th} row of the matrix U are denoted by U_{i*}. Therefore, the loss function of the HTM for tensor factorization is defined by

$$\begin{aligned} F(\mathcal{H}, \mathcal{S}, U, I, T) \quad &= 1/2 \|\mathcal{S} \times_U U \times_I I \times_C C - \mathcal{H}\|_F^2 \\ &+ 1/2 \left[\lambda_U \|U\|_F^2 + \lambda_I \|I\|_F^2 + \lambda_C \|C\|_F^2\right], \end{aligned} \tag{3}$$

where $\mathcal{S} \in \mathbb{R}^{d_U \times d_I \times d_C}$ represents a central tensor; $\| \cdot \|_F^2$ indicates the Frobenius norm; $U \in \mathbb{R}^{I \times d_U}$, $I \in \mathbb{R}^{J \times d_I}$, and $C \in \mathbb{R}^{K \times d_C}$ are matrices of users, items, and criteria; d_U, d_I, and d_C are parameters adjusting the dimensionality of latent factors; λ_U, λ_I, and λ_C are the regularization parameters.

Because of the absence of a closed-form solution for the minimization of Eq. (3), the loss function is minimized by Stochastic Gradient Descent (SGD). Algorithm 1 shows the procedures of tensor factorization by using Higher Order Singular Value Decomposition (HOSVD) [13] for the proposed HTM, where the gradients of our objective function can be calculated as follows:

$$\begin{aligned}
\eta\partial_{U_{i*}}F^L &= (\hat{\mathcal{H}}_{ijk} - \mathcal{H}_{ijk}) \times \mathcal{S} \times_I I_{j*} \times_C C_{k*},\\
\eta\partial_{I_{j*}}F^L &= (\hat{\mathcal{H}}_{ijk} - \mathcal{H}_{ijk}) \times \mathcal{S} \times_U U_{i*} \times_C C_{k*}, \text{ and}\\
\eta\partial_{C_{k*}}F^L &= (\hat{\mathcal{H}}_{ijk} - \mathcal{H}_{ijk}) \times \mathcal{S} \times_U U_{i*} \times_I I_{j*}.
\end{aligned} \tag{4}$$

This algorithm linearly scales to the number of rating values R and iteration number L

Algorithm 1: Factorization of hypothetical tensor model

h **Data:** observed tensor $\mathcal{H}$, learning rate t_0, torelance tol, maxEpoch $maxEpo$
regularization parameters $\lambda = \lambda_U = \lambda_R = \lambda_C$
Result: approximate tensor $\hat{\mathcal{H}}$

Initialize $\mathcal{H}, \mathcal{S}, U, I, C$ with zero ;
Set $l = 0$, $t = t_0$, $tol = 5$, and $maxEpo = 10$;
while *not converged and* $l < maxEpo$ **do**
    $\eta = 1/\sqrt{t}$ and $t = t + 1$;
    for *each* $\mathcal{H}_{ijk} \neq 0$ **do**
        Update the U_{i*}, I_{j*}, and C_{k*} by Eq. (4) ;
        Compute the objective function F^L by Eq. (3) ;
    end
    Compute training loss tl_l in l^{th} iteration ;
    Compute change rate $cRate = (tl_{l-1} - tl_l)/tl_{l-1} * 100$;
    if $cRate < tol$ **then**
        break;
    end
    $l = l + 1$;
end
Return $\hat{\mathcal{H}} = \mathcal{S} \times_U U \times_R R \times_C C$;

and the dimensionalities I, J, and K of user, item and criterion factors. Therefore, a time complexity of the proposed algorithm is $\mathcal{O}(LRIJK)$. It is worth to mention that tensor factorization of our models are faster than conventional ones because the R and K are constants and the iteration number is less than $L = 10$. Therefore, the final complexity of our approach thus is $\mathcal{O}(FIJ)$ with constant $F \leqq 10 \times RK$ actually.

4. Evaluation protocol and metrics

4.1. Dataset and data analysis

The dataset used in our experiments contains 44,217 multiple ratings of 19,970 users to 2,484 restaurants gathered from Tripadvisor. Note that we used all data in the dataset without any filtering in order to consider new users with partial preferences. Since this study considers four rating criteria, sparsities of the proposed models is higher than other compared methods based on a user-item matrix. For example, if the tensor model consists of the users, restaurants, and multiple ratings, its sparsity is very high, as 99.978% (density is 0.022%). According to [26], this sparsity is natural in real-world situations but hampers the accuracy of recommendation systems.

Indeed, let's look at the dataset in details to briefly discuss how many unknown ratings are collected in terms of MCRSs. Table 1 presents the distribution of multiple ratings in the dataset. The percentage expression in brackets indicates a ratio of rating number divided by the total number, and the ratios in the last row are between the total numbers of the overall type and the other rating types. Note that we also found that there are some users, who input only other ratings without overall one (i.e., the second pattern), but they are very few (around 89) in the dataset. Therefore, we can glance at the problem of new users with partial preferences in the dataset via ratios and distributions showed in the table above. Brief but meaningful analysis results are as follows:

Table 1. Rating distribution for multi-criteria

Rating	Overall (%)	Food (%)	Price (%)	Service (%)
1	1,027 (5.96)	485 (5.49)	640 (7.09)	641 (7.02)
2	1,144 (6.64)	532 (6.02)	700 (7.76)	505 (5.53)
3	2,122 (12.32)	1,113 (12.59)	1,597 (17.70)	1,103 (12.08)
4	5,287 (30.70)	2,562 (28.98)	2,876 (31.87)	2,436 (26.68)
5	7,642 (44.37)	4,148 (46.92)	3,210 (35.58)	4,446 (48.69)
Total	17,222	8,840 (51.33)	9,023 (52.39)	9,131 (53.02)

– In terms of rating types, the overall rating is usually filled by most of the users, while half of the other types have the unknown values. In the context of MCRSs, it emphasizes why the unknown class patterns caused by users' unfamiliarity need to be addressed. It will be discussed more in Section 5.2.
– There are much more positive ratings (around 75 %) than negative ones, and it is important to analyze the effects of proposed rules, especially $\mathcal{R}_1$ and $\mathcal{R}_3$. In other words, the effectiveness of $\mathcal{R}_2$ could be relatively decreased because of the small number of low ratings.

On the other hand, if a multi-dimensional model such as matrix or tensor maintains inherent relations between multiple types of ratings, correlations between them are also significant for MCRSs. Table 2 shows the correlations between rating types in two datasets. One consists of fully filled data only, and the other fills empty values by zero as all data to compute correlations. For the fully filled data, each criterion shows high correlations with the others. In particular, all other criteria have higher correlations with overall criterion than 0.79. However, when we consider all data without any complementary ways, the correlations are decreased as around 0.2. It will significantly and negatively affect the performance of recommender systems based on the multi-dimensional models. Therefore, we need to carefully deal with new users with partial preferences in such systems. One other interesting is that the correlations between other rating types except for overall type are higher than 0.92, in the case of all data. It is because of that half of the other rating

Table 2. Correlation between multi-criteria

	Fullly filled data (8,612)				All data (17,222)			
	Overall	Food	Price	Service	Overall	Food	Price	Service
Overall	1.000	0.873	0.839	0.794	1.000	0.197	0.207	0.181
Food	0.873	1.000	0.813	0.687	0.197	1.000	0.941	0.926
Price	0.839	0.813	1.000	0.693	0.207	0.941	1.000	0.938
Service	0.794	0.687	0.693	1.000	0.181	0.926	0.938	1.000

types are empty and filled by zero. It emphasizes that a model must distinguish between unknown and lowest ratings to avoid this kind of bias. In this paper, we only used known ratings and unknown ratings complemented by the proposed rules to train models.

4.2. Experimental protocol

To compare prediction performances of the proposed method and other techniques, three measures (i.e., Root Mean Square Error (RMSE), Mean Square Error (MSE), and Mean Absolute Error (MAE)) are exploited as follows:

$$RMSE = \sqrt{\sum_{i=1}^{N} (\hat{r}_i - r_i)^2 / N}, \tag{5}$$

$$MSE = \sum_{i=1}^{N} (\hat{r}_i - r_i)^2 / N, and \tag{6}$$

$$MAE = \sum_{i=1}^{N} |\hat{r}_i - r_i| / N, \tag{7}$$

where $\hat{a}_i$ and a_i are predicted and observed rating scores, respectively. The N indicates the number of compared ratings.

We use k-fold cross-validation scheme to compare errors between predicted and observed ratings with avoiding overfitting impacts in all the following experiments. In this regard, it is significant to select a proper value for k since a poorly chosen k value could cause a misrepresentation (e.g., overestimation or high variance). According to "Typically, $k = 5$ or $k = 10$ have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor very high variance [16]," we set the k as 5.

4.3. Baseline Models

This section explains other baselines. The two-dimensional techniques using user-item matrix and basic MCRSs based on the techniques are as follows:

K-Nearest Neighbors-based CF (KNN): is one of the basic CF algorithms. The predicted value $\hat{r}_{ui}$ of user u to item i is defined by $\hat{r}_{ui} = \sum_{v \in \sum_i^n (u)} sim(u, v) \cdot r_{vi} / \sum_{v \in \sum_i^k (u)} sim(u, v)$, where n is the number of neighbors and $sim()$ denotes a similarity function.

KNN BaseLine-based CF (KNN-BL): inspired by [15] models neighborhood relations by minimizing a global cost function. The prediction $\hat{r}_{ui}$ is defined as $\hat{r}_{ui} =$

$b_{u,i}+\sum_{v\in\sum_{N_i^k(u)}} sim(u,v)\cdot(r_{vi}-b_{vi})/\sum_{v\in\sum_{N_i^k(u)}} sim(u,v)$, where $b_{u,i}=\mu+b_u+b_i$ indicates baseline estimates with an overall average rating μ and observed deviations b_u and b_i of users and items. The n and $sim()$ are equal to those of KNN.

Co-clustering-based CF (COC) [8]: assigns users and items into some user and item clusters. Rating scores of a target user are then predicted based on ratings of users and items belonging to the same cluster of the target user. An approximate rating $\hat{r}_{ui}$ is computed as follows: $\hat{r}_{ui}=\overline{C_{ui}}+(\eta_u-\overline{C_u})+(\eta_i-\overline{C_i})$, where the $\overline{C_{ui}}$, $\overline{C_u}$ and $\overline{C_i}$ are average ratings of co-cluster C_{ui}, u's cluster and i's cluster, respectively.

Singular Value Decomposition-based MF (SVD): has been popularized by Simon Funk during the Netflix Prize. The prediction $\hat{r}_{ui}$ is set as: $\hat{r}_{ui}=\eta+b_u+b_i+q_i^T p_u$. If user u or item i is unknown, then the biases b_u or b_i and the factors p_u or q_i are assumed to be zero. To estimate all unknown values, a loss function is usually minimized by stochastic gradient descent.

SVD++-based MF (SVD++) [14]: is an extension of the SVD considering implicit ratings. The prediction $\hat{r}_{ui}$ is calculated by $\hat{r}_{ui}=\eta+b_u+b_i+q_i^T*(p_u+|I_u|^{-1/2}\sum_{j\in I_u} y_j)$, where the y_j term indicates a new set of item factors capturing implicit ratings. Also, the implicit rating means that a user u rated an item j, regardless of the rating value.

Non-negative Matrix Factorization (NMF) [20]: is similar to SVD. The predicted rating $\hat{r}_{ui}$ is set as: $\hat{r}_{ui}=q_i^T p_u$, where user and item factors keep positive. For optimization, a (regularized) stochastic gradient descent is used.

Aggregation-based Multi-criteria Recommendation (AMR): On the other hand, we also compare our method with conventional MCRS methods that has been proposed in [1]. Among the MCRS techniques, we implemented aggregation-function-based approach based on the above-mentioned two-dimensional techniques. Such approaches have three steps. The first stage predicts missing rating values via a single-criterion method based on the user-item matrix for each criterion. Second stage aims to estimate relationships between the overall rating and other criteria, so we used linear, Ridge, and Lasso regressions in order to get best coefficients between overall rating with the others (i.e., food, price, and service ratings). Lastly, the approximated values are aggregated into overall ratings with the coefficients obtained as weights in this aggregation stage.

We implemented the two-dimensional recommendation techniques by using surprise library and conducted a grid-search to find optimal parameters. The AMR was developed by using the techniques and regression methods of Scikit-learn library. All the experiments including parallel parameter searches conducted in the same computation environment consisting of 40 CPUs and RAM 128GB.

5. Evaluation and discussion

This section compares the proposed Hypothetical Tensor Model (HTM) with other baseline methods and discusses effectiveness of the proposed rules to alleviate the problem of new users with partial preferences in the context of MCRSs.

5.1. Performance comparison

Figure 3 shows the RMSE and MAE of two-dimensional techniques and HTM. The compared methods have much lower performances than HTM in terms of both measures. The

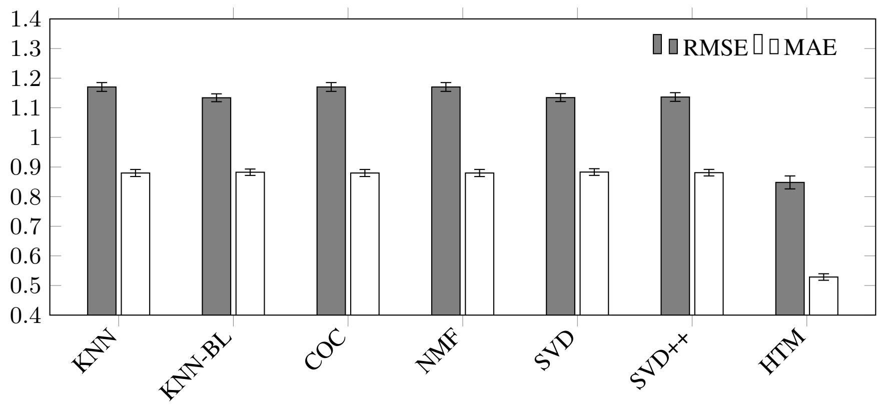

Fig. 3. Performance comparison with two dimensional-based techniques

two-dimensional methods for the dataset including new users with partial ratings are average RMSE 1.1525 and MAE 0.8810. It means that they are negatively affected by the new users as one of cold-start problems. However, the proposed HTM outperform them, with average RMSE 0.8480 and MAE 0.5286. These results support that our proposed model can improve the recommendation performance and alleviate the new user problem.

The HTM is also compared with basic MCRSs (i.e., AMRs). Figure 4 presents performances of the various AMRs with different two-dimensional techniques and HTM. Experimental results show that the proposed HTM significantly outperforms than the AMRs. Because the AMRs are based on the two-dimensional methods, and such methods show low performances on the dataset including new users, the performance of AMRs are also affected negatively. Even the AMRs with SVD and SVD++ show bad performances than two-dimensional methods SVD and SVD++. It emphasizes that proper handling of new users with partial preferences is significantly important to MCRSs.

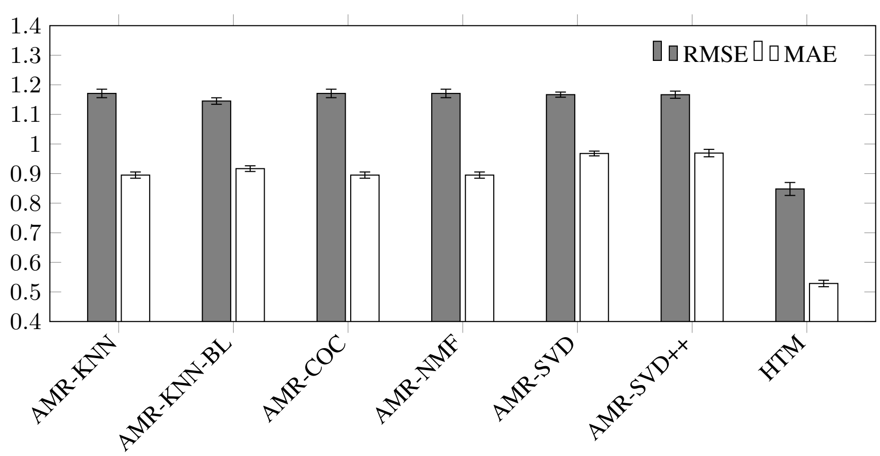

Fig. 4. Performance comparison with basic MCRS techniques

To more discussion, we selected KNN-BL, COC, SVDpp, and ARMs based on them, because they show better performance than the other two-dimensional techniques. Table 3 lists performances of all the methods and MSE ratios of HTM to the other techniques. We use the MSE ratio to see how much is a better performance of the proposed method than baseline methods. The ratio is defined by $(1 - MSE_{targ}/MSE_{comp}) \times 100$, where the MSE_{targ} and MSE_{comp} indicate MSE errors of target and compared methods. Note that all the techniques were tested with the same number 3,344 of test samples with 5-fold cross-validation to avoid some bias which can happen by different sizes of test set. If the ratio is a positive value 50, it means that a target method is better as more 50 percentage than a compared one. As a result, the HTM improves the minimum 40% (maximum 47%)

Table 3. Comparison of HTM with other techniques

Measure		two-dimensional techniques			ARMs			HTM
		KNN-BL	COC	SVD++	KNN-BL	COC	SVD++	
Mean	MSE	1.2855	1.3693	1.2912	1.3114	1.3711	1.3615	**0.7192**
	RMSE	1.1338	1.1702	1.1363	1.1452	1.1709	1.1668	**0.8480**
	MAE	0.8824	0.8798	0.8809	0.9167	0.8949	0.9679	**0.5286**
Std	MSE	0.0302	0.0348	0.0333	0.0252	0.0333	**0.0203**	0.0372
	RMSE	0.0133	0.0149	0.0147	0.0110	0.0143	**0.0087**	0.0219
	MAE	0.0108	0.0119	0.0110	0.0096	0.0105	**0.0081**	0.0110
MSE comparison (%)		44.06	47.48	44.30	45.16	47.55	47.18	

than the baseline methods in terms of the MSE measure, see the last row. In general, the more deviations between RMSE and MAE, the more substantial error variance. It means that errors of predicted ratings are unstable. In this regard, HTM shows slightly higher standard deviations of MSE, RMSE and MAE than the ARMs, while it has similar ones with two-dimensional methods. Therefore, HTM's instability on the preference prediction is acceptable.

5.2. Effectiveness of unknown class rules

This section debates the efficacy of unknown class rules. Table 4 lists performances of ARMs with the above three two-dimensional techniques and the proposed rules. The notation RARM indicates ARM based on the proposed rules. As a result, RARMs based

Table 4. Effectiveness comparison of AMRs by proposed rules

Measure		KNN-BL		COC		SVD++	
		RARM	ARM	RARM	ARM	RARM	ARM
Mean	RMSE	1.1374	1.1452	1.1814	1.1709	1.1429	1.1666
	MAE	0.8729	0.9167	0.8980	0.8949	0.8675	0.9693
Std	RMSE	0.0152	0.0110	0.0174	0.0143	0.0181	0.0121
	MAE	0.0113	0.0096	0.0122	0.0105	0.0122	0.0126

on KNN-BL and SVD++ have better performance than the ARMs, except for COC base.

Moreover, the two methods show slightly low RMSE and MAE errors than two-dimensional techniques (see KNN-BL and SVD++ in Table 3) despite the standard deviation of them is similar. Thus, experimental results support that there is the potential effectiveness for the basic MCRSs. However, there were no high improvements, since it might be difficult to effectively keep inherent relationships between multi-criteria via such methods based two-dimensional matrix.

Table 5 shows performances of the HTM for each multi-criterion and all ratings by the proposed rules. Bold and underline font styles represent first and second-best performances, respectively. The CTF denotes a conventional tensor factorization that isn't applied by the proposed rules. Performances of the HTM with all rules is marked by italic font style. To fairly compare between model performances, optimal parameters for CTF and HTMs were equally set as regularization parameter $\lambda = 0.01$, learning rate $t_o = 0.001$, latent factors '3-2-2' of users, items, and rating types.

Table 5. Effectiveness comparison of HTM for multi-criteria

		Mean					Std				
		CTF	$\mathcal{R}_1$	$\mathcal{R}_2$	$\mathcal{R}_3$	HTM	CTF	$\mathcal{R}_1$	$\mathcal{R}_2$	$\mathcal{R}_3$	HTM
All	MSE	1.0412	**0.6376**	1.0355	1.0368	*0.6158*	0.0251	**0.0063**	0.0270	0.0155	*0.0142*
	RMSE	1.0203	**0.7985**	1.0175	1.0182	*0.7847*	0.0123	**0.0039**	0.0132	0.0076	*0.0091*
	MAE	0.6790	**0.5307**	0.6763	0.6772	*0.5221*	0.0074	**0.0048**	0.0070	0.0048	*0.0055*
Overall	MSE	1.8015	**0.7370**	1.7979	1.8181	*0.7196*	0.0449	**0.0129**	0.0487	0.0442	*0.0372*
	RMSE	1.3421	**0.8585**	1.3407	1.3483	*0.8480*	0.0168	**0.0075**	0.0181	0.0163	*0.0219*
	MAE	0.9237	**0.5403**	0.9205	0.9248	*0.5286*	0.0082	**0.0054**	0.0133	0.0112	*0.0110*
Food	MSE	0.4561	0.4723	0.4493	**0.4396**	*0.4453*	0.0246	0.0279	**0.0172**	0.0234	*0.0265*
	RMSE	0.6751	0.6869	0.6702	**0.6628**	*0.6670*	0.0181	0.0204	**0.0128**	0.0180	*0.0199*
	MAE	0.4535	0.4667	0.4511	**0.4472**	*0.4576*	0.0111	0.0104	0.0115	**0.0089**	*0.0104*
Price	MSE	0.4994	0.5148	0.4948	**0.4825**	*0.4939*	0.0162	0.0290	0.0187	**0.0120**	*0.0044*
	RMSE	0.7066	0.7172	0.7033	**0.6946**	*0.7028*	0.0115	0.0205	0.0133	**0.0086**	*0.0032*
	MAE	0.5421	**0.5279**	0.5420	0.5371	*0.5247*	**0.0035**	0.0104	0.0062	0.0062	*0.0030*
Service	MSE	0.6945	0.7129	0.6884	**0.6834**	*0.6876*	**0.0242**	0.0451	0.0321	0.0297	*0.0332*
	RMSE	0.8333	0.8439	0.8295	**0.8265**	*0.8290*	**0.0145**	0.0269	0.0193	0.0180	*0.0199*
	MAE	0.5648	0.5751	**0.5600**	0.5623	*0.5664*	**0.0101**	0.0157	0.0145	0.0116	*0.0106*

In terms of rating types (i.e., each criterion), $\mathcal{R}_1$ positively affects the "Overall" rating, and $\mathcal{R}_2$ and $\mathcal{R}_3$ show their effectiveness to other extra rating types (i.e., "Food", "Price", and "Service"). However, the magnitudes of their impacts differ. As we glanced in Table 1, it is because of the different numbers of data that can be applied by each rule. Indeed, rating numbers of data applied by $\mathcal{R}_1$, $\mathcal{R}_2$, and $\mathcal{R}_3$ were $4,345$, $1,582$, and $2,371$. In this regard, three rules have acceptable efficiencies to improve the performance of HTM.

It is worthy to mention that although the CTF for dataset including new users had worse performance than two-dimensional methods (see only results from "Overall" rating to compare fairly), other extra rating types showed improved performances by comparing with the "Overall" rating. As a result, performances of CTF for all data show improvements than the two-dimensional techniques (see the row "All") in terms of both mean and standard deviation of MSE, RMSE, and MAE. It means that representation via tensor models can improve stably performance because of their characteristics maintaining an inherent structure of and relationships between multi-criteria.

On the other hand, our HTM outperform than the other techniques including CTF in terms of “Overall” rating types. Furthermore, the standard deviations of RMSE and MAE of HTM for most of the rating types (except for “Service”) are similar to or smaller than the CTF. Thus, HTM has better stability to predict user preferences than conventional tensor factorization methods. Furthermore, HTM show 40.86% performance improvement from the CTF by MSE comparison for “All” ratings. As a result, experimental results verified that handling of the problem new users in MCRSs is one of significant tasks, and the HTM solves the problem well.

Lastly, we discuss correlations between multi-criteria by the proposed rules ($\mathcal{RU}()$). As afore-discussed, there are two kinds of dataset (i.e., fully filled data and all data) to compare correlations. Figure 5 shows the correlation heat-maps between multi-criteria of original data and data complemented by the proposed rules. For the ‘fully filled data’ of

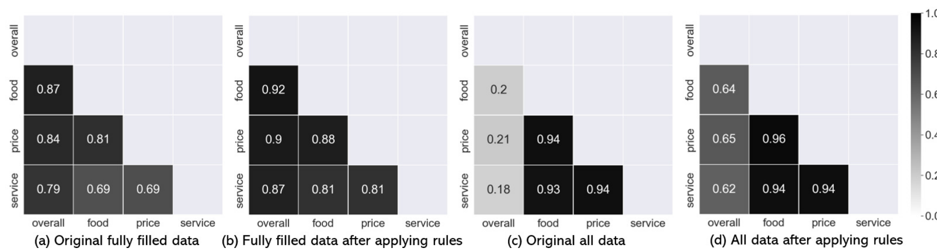

Fig. 5. Correlation between multi-criteria with rules

(a) and (b) in the figure, $\mathcal{RU}()$ increases correlations between multiple ratings. Even, for the ‘all data’ including new users, overall rating’s correlations with the other ratings are increased than original all data (see (c) and (d) in the figure above). It would be the reason why our HTM outperforms the CTF.

6. Conclusion

MCRSs have been developed to improve the accuracy of traditional single-criterion recommender systems that cannot express and reflect fine-grained users’ rating behaviors. However, in some MCRSs, new users would be asked to express their preferences on some criteria of items in order to address the cold-start problem. Such collected preferences are often incomplete because of the users’ unfamiliarity on the rating scheme and recommendation domain, which are called ‘partial preferences’. The issue of new users as one of the cold-start problems thus decrease the accuracy performance of MCRSs.

To address the problem in the context of MCRSs, we found four patterns of partial preferences that are caused by new users’ unfamiliarity via data analysis. And then, an intuitive rule set based on relationships between multi-criteria is defined to alleviate the negative impact of partial preferences. As a result, the Hypothetical Tensor Model (HTM) based on the rules is proposed to maintain a structure of and correlations between multi-criteria and to improve the performance of MCRSs. Experiments on a TripAdvisor dataset showed the better performances of the HTM than baseline methods as well as the effectiveness of the proposed rules for each criterion.

Since there are still some limitations in the proposed model, we plan two future pieces of research. One relates to $\mathcal{R}_1$ which is currently applied to the minimum or maximum rating. As the rule can be applied many data than the other rules, we will find an appropriate machine learning technique to complete other values of ratings. The other is relevant to the high computational cost of the proposed method. Even though, as afore-discussed in Section 3.3, some parts of computational complexity are constant, but the cost is still high. Therefore, we will leverage a clustering method to reduce the computational cost.

Acknowledgement. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2020R1A2B5B01002207).

References

1. Adomavicius, G., Kwon, Y.: New recommendation techniques for multicriteria rating systems. IEEE Intelligent Systems 22(3), 48–55 (2007)
2. Al-Ghuribi, S.M., Noah, S.A.M.: Multi-criteria review-based recommender system-the state of the art. IEEE Access 7, 169446–169468 (2019)
3. Borràs, J., Moreno, A., Valls, A.: Intelligent tourism recommender systems: A survey. Expert Systems with Applications 41(16), 7370–7389 (2014)
4. Ebadi, A., Krzyzak, A.: A hybrid multi-criteria hotel recommender system using explicit and implicit feedbacks. International Journal of Computer and Information Engineering 10(8), 1377–1385 (2016)
5. Farokhi, N., Vahid, M., Nilashi, M., Ibrahim, O.: A multi-criteria recommender system for tourism using fuzzy approach. Journal of Soft Computing and Decision Support Systems 3(4), 19–29 (2016)
6. Fu, Y., Liu, B., Ge, Y., Yao, Z., Xiong, H.: User preference learning with multiple information fusion for restaurant recommendation. In: Zaki, M.J., Obradovic, Z., Tan, P., Banerjee, A., Kamath, C., Parthasarathy, S. (eds.) In Proceedings of the 2014 SIAM International Conference on Data Mining. pp. 470–478. SIAM, Philadelphia, Pennsylvania, USA (Apr 2014)
7. Gavalas, D., Konstantopoulos, C., Mastakas, K., Pantziou, G.E.: Mobile recommender systems in tourism. Journal of Network and Computer Applications 39, 319–333 (2014)
8. George, T., Merugu, S.: A scalable collaborative filtering framework based on co-clustering. In: In Proceedings of the 5th IEEE International Conference on Data Mining (ICDM 2005). pp. 625–628. IEEE Computer Society, Houston, Texas, USA (Nov 2005)
9. Hassan, M., Hamada, M.: A neural networks approach for improving the accuracy of multi-criteria recommender systems. Applied Sciences 7(9), 868 (2017)
10. Hassan, M., Hamada, M.: Genetic algorithm approaches for improving prediction accuracy of multi-criteria recommender systems. International Journal of Computational Intelligence Systems 11(1), 146–162 (2018)
11. Hong, M., Akerkar, R., Jung, J.J.: Improving explainability of recommendation system by multi-sided tensor factorization. Cybernetics and Systems 50(2), 97–117 (2019)
12. Hong, M., Jung, J.J.: Multi-sided recommendation based on social tensor factorization. Information Sciences 447, 140–156 (2018)
13. Karatzoglou, A., Amatriain, X., Baltrunas, L., Oliver, N.: Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In: Amatriain, X., Torrens, M., Resnick, P., Zanker, M. (eds.) In Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010. pp. 79–86. ACM, Barcelona, Spain (Sep 2010)

14. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Li, Y., Liu, B., Sarawagi, S. (eds.) In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 426–434. ACM, Las Vegas, Nevada, USA (Aug 2008)
15. Koren, Y.: Factor in the neighbors: Scalable and accurate collaborative filtering. ACM Transactions on Knowledge Discovery from Data 4(1), 1:1–1:24 (2010)
16. Kuhn, M., Johnson, K.: Applied predictive modeling, vol. 26. Springer (2013)
17. Kumar, G.: A survey on multi criteria decision making recommendation system using sentiment analysis. International Journal of Applied Engineering Research 13(15), 11724–11729 (2018)
18. Lakiotaki, K., Matsatsinis, N.F., Tsoukiàs, A.: Multicriteria user modeling in recommender systems. IEEE Intelligent Systems 26(2), 64–76 (2011)
19. Lu, J., Wu, D., Mao, M., Wang, W., Zhang, G.: Recommender system application developments: A survey. Decision Support Systems 74, 12–32 (2015)
20. Luo, X., Zhou, M., Xia, Y., Zhu, Q.: An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. IEEE Transactions on Industrial Informatics 10(2), 1273–1284 (2014)
21. Musto, C., de Gemmis, M., Semeraro, G., Lops, P.: A multi-criteria recommender system exploiting aspect-based sentiment analysis of users' reviews. In: Cremonesi, P., Ricci, F., Berkovsky, S., Tuzhilin, A. (eds.) In Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017. pp. 321–325. ACM, Como, Italy (Aug 2017)
22. Nassar, N., Jafar, A., Rahhal, Y.: A novel deep multi-criteria collaborative filtering model for recommendation system. Knowledge-Based Systems 187 (2020)
23. Pappas, N., Popescu-Belis, A.: Adaptive sentiment-aware one-class collaborative filtering. Expert Systems with Applications 43, 23–41 (2016)
24. Rendle, S., Marinho, L.B., Nanopoulos, A., Schmidt-Thieme, L.: Learning optimal ranking with tensor factorization for tag recommendation. In: IV, J.F.E., Fogelman-Soulié, F., Flach, P.A., Zaki, M.J. (eds.) In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 727–736. ACM, Paris, France (Jun 2009)
25. Sahoo, N., Krishnan, R., Duncan, G.T., Callan, J.: Research note - the halo effect in multicomponent ratings and its implications for recommender systems: The case of yahoo! movies. Information Systems Research 23(1), 231–246 (2012)
26. Singh, M.: Scalability and sparsity issues in recommender datasets: a survey. Knowledge and Information Systems pp. 1–43 (2018)
27. Wasid, M., Ali, R.: An improved recommender system based on multi-criteria clustering approach. Procedia Computer Science 131, 93–101 (2018)
28. Zhang, S., Salehan, M., Leung, A., Cabral, I., Aghakhani, N.: A recommender system for cultural restaurants based on review factors and review sentiment. In: In Proceedings of the 24th Americas Conference on Information Systems, AMCIS 2018. New Orleans, LA, USA (Aug 2018)
29. Zheng, X., Luo, Y., Sun, L., Zhang, J., Chen, F.: A tourism destination recommender system using users' sentiment and temporal dynamics. Journal of Intelligent Information Systems 51(3), 557–578 (2018)
30. Zheng, Y.: Criteria chains: A novel multi-criteria recommendation approach. In: Papadopoulos, G.A., Kuflik, T., Chen, F., Duarte, C., Fu, W. (eds.) In Proceedings of the 22nd International Conference on Intelligent User Interfaces, IUI 2017. pp. 29–33. ACM, Limassol, Cyprus (Mar 2017)
31. Zhongqin, B., Shuming, D., Zhe, L., Yongbin, L.: A recommendations model with multiaspect awareness and hierarchical user-product attention mechanisms. Computer Science and Information Systems 17(3), 849-865 (2020)

Minsung Hong as a postdoctoral researcher in Western Norway Research Institue, Norway, since February 2018 is participating in several EU Horizon2020 and Norway national projects. He received the B.S. and M.S. degrees in Computer Engineering from DanKook University in 2011 and 2014, respectively. He received the Ph.D. degree in Computer Engineering from Chung-Ang University in 2018. His research topics are recommender systems based on big data by using various the methodologies of artificial intelligence, data mining, and machine learning.

Jason J. Jung is a Full Professor in Chung-Ang University, Korea, since September 2014. Before joining CAU, he was an Assistant Professor in Yeungnam University, Korea since 2007. Also, he was a postdoctoral researcher in INRIA Rhone-Alpes, France in 2006, and a visiting scientist in Fraunhofer Institute (FIRST) in Berlin, Germany in 2004. He received the B.Eng. in Computer Science and Mechanical Engineering from Inha University in 1999. He received M.S. and Ph.D. degrees in Computer and Information Engineering from Inha University in 2002 and 2005, respectively. His research topics are knowledge engineering on social networks by using many types of AI methodologies, e.g., data mining, machine learning, and logical reasoning. Recently, he have been working on intelligent schemes to understand various social dynamics in large scale social media.

Received: May 31, 2020; Accepted: November 12, 2020.