Intelligent Data Analysis 21 (2017) S195–S216
DOI 10.3233/IDA-170878
IOS Press

Effective social content-based collaborative filtering for music recommendation

Ja-Hwung Su[a,*], Wei-Yi Chang[b] and Vincent S. Tseng[c]
[a]*Department of Information Management, Cheng Shiu University, Kaohsiung, Taiwan*
[b]*Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan*
[c]*Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan*

Abstract. Recently, music recommender systems have been proposed to help users obtain the interested music. Traditional recommender systems making attempts to discover users' musical preferences by ratings always suffer from problems of rating diversity, rating sparsity and lack of ratings. These problems result in unsatisfactory recommendation results. To deal with traditional problems, in this paper, we propose a novel music recommender system, namely Multi-modal Music Recommender system (MMR), which integrates social and collaborative information to predict users' preferences. In this work, the playcounts are transformed into collaborative information to cope with problem of lack of rating information, while item tags and artist tags are employed as social information to cope with problems of rating diversity and rating sparsity. Through optimizing the integrated social-and-collaborative information, the users' preferences can be inferred more accurately and efficiently. The experimental results reveal that, three problems can be alleviated significantly and our proposed method outperforms other state-of-the-art recommender systems in terms of RMSE (Root Mean Square Error) and NDCG (Normalized Discount Cumulative Gain).

Keywords: Music recommendation, collaborative filtering, social content, data engineering, nonnegative matrix factorization

1. Introduction

Recent advances in multimedia technology make music data available at an explosive rate. It incurs a problem that, it is not easy to efficiently and effectively acquire the preferred music pieces (called items in this paper) from a massive amount of music data. To deal with this problem, a large increase in needs of music recommendation is enabled then. So-called music recommendation refers to a set of predictive algorithms learning musical preferences from users' usage logs. Hence, identifying the preferences in users' usage logs is a challenging issue for a recommender system. Typically, users' preferences are described by explicit votes and implicit votes. Explicit votes describe a user's preferences using a numerical rating, while implicit votes describe the preferences using play counts, tags, comments, listening history and other types of usage information. Based on the votes, the well-known Collaborative Filtering (CF) is proposed to predict users' musical preferences on.

For CF using explicit votes, the users' ratings are collected into a user-to-item rating matrix. The user-to-item rating matrix contains elements that range from zero to five, where zero indicates that the items are never rated by the users. In addition to zero elements, users present their preferences by values from

*Corresponding author: Ja-Hwung Su, Department of Information Management, Cheng Shiu University, Kaohsiung, Taiwan. E-mail: bb0820@ms22.hinet.net.

Table 1
Example of a user-to-item rating matrix

	Item 1	Item 2	Item 3	Item 4
user 1	0	0	0	0
user 2	***4***	4	0	3
user 3	0	***1***	0	5
user 4	1	4	0	3

one to five. According to the rating matrix, the recommendation procedure for this type of CF is divided into two stages, namely rating prediction and item selection.

I. Rating prediction stage: The goal of this stage is to predict the ratings for un-purchased items. The un-purchased items stand for those that have never been rated by the active user.
II. Item selection stage: When the ratings of un-purchased items are predicted, the ranking list for un-purchased items is derived. Finally, the top N items are recommended to the active user.

In general, most past studies have focused their attention on the first stage because the second stage is straightforward and cheaper. Although this type of CF has been shown to be effective, there remain ms, which are addressed as follows.

1.1. Rating diversity

The rating-diversity indicates that, the ratings are inconsistent among items or users. If there are inconsistent ratings, the rating prediction methods cannot generate good results. Table 1 is an example of this issue. In Table 1, there are four users and four items in the user-to-item rating matrix. Assume the 1st item of user 2 and the 2nd item of user 3 are the target items to predict for user-based CF [8,17] and item-based CF [7,21], respectively. The most relevant item to item 2 for item-based CF is item 4 and the most relevant user to user 2 for user-based CF is user 4. Therefore, the rating of item 2 for user 3 is predicted to be 5, while that of item 1 for user 2 is 1. Obviously, the prediction errors, which are $5 - 1 = 4$ and $4 - 1 = 3$ for item-based CF and user-based CF, respectively, are too large. This example shows that, the large gap between the predictions and users' preferences are often caused by inconsistent ratings.

1.2. Rating sparsity

Another performance bottleneck of recommender systems based on ratings is rating sparsity. The rating-sparsity indicates that, there are a large number of zero elements in the user-to-item rating matrix. For CF, a sparse matrix gives unreliable prediction results since there is insufficient information. Table 1 is a proper example depicting that, it is difficult to predict the 1st user's preferences accurately since her/his ratings are too few. Also, the ratings of the 3rd item cannot be predicted successfully. Unfortunately, this problem occurs frequently in real applications.

1.3. Lack of rating information

In real applications, users listen to music and give tags to music without ratings. For this type of music recommender systems, it is because no explicit rating is employed to describe the user's preferences that the music preference is hard to predict. This problem is called "lack of rating information" in this paper.

Due to the lack of rating information, another type of CF predicts the users' preferences using implicit votes such as social media hits. Consequently, studies increasingly use social media information to determine users' preferences instead of traditional ratings. The preference prediction type then does not use ratings, but probabilistic user-to-item relationships. In other words, this type of recommendation mechanisms predicts the relevance between the users and items, and thereby generates a ranking list of items (so-called Top-N Recommendation, TNR). Although social media information has been shown to be relevant to the users' preferences [11,19,25], the predicted user-to-item relevance is not robust enough to determine the real preference for an item or music.

In this paper, we propose a novel music recommender system called Multi-modal Music Recommender system (MMR), which achieves high quality of music recommendation and addresses the problems defined. First, the playcounts are transformed into ratings to deal with the problem of the lack of ratings. Second, the tag information and rating information are integrated to deal with problems of rating diversity and rating sparsity. Third, the integrated information is further optimized to reduce the computational cost and to discover the important factors by performing nonnegative matrix factorization algorithm. Fourth, the artist information and optimized information are fused to improve the prediction quality. The experimental results reveal that, the proposed MMR performs better than existing methods in terms of RMSE (Root Mean Square Error) and NDCG (Normalized Discount Cumulative Gain).

The remainder of this paper is organized as follows. A brief review of related works is given in Section 2. Section 3 describes the proposed approach for predicting users' preferences. Empirical evaluations of our approach are expressed in Section 4. Finally, the conclusions and future works are given in Section 5.

2. Related work

The goal of music recommendation is to help the user make a decision for choosing the preferred music pieces from a large music archive by associating the users' preferences with music. To this end, a considerable number of past studies have been made on music recommendation. Although these forerunners have been shown to be effective, there remain some problems unsettled. In the followings, the review of past studies is briefly described by categories.

2.1. Memory-based CF

This is a traditional recommendation paradigm that infers the ratings by a user-to-item rating matrix. It is well known that user-based recommender systems [8,17] predict the item ratings by the most-relevant users on similar ratings, while item-based ones [7,21] predict the item ratings by the most-relevant items on similar ratings. In order to attack the individual lacks of above methods, Wang et al. [26] proposed an algorithm to unify the user-based and item-based CFs. Another user-based CF using significances of the users and items to predict the users' preferences is proposed by Bobadilla et al. [2]. In fact, the significances in [2] are still calculated by ratings so that it cannot represent the users' preferences precisely. As a whole, this type of recommender systems considering only ratings encounters problems of rating-diversity and rating sparsity mentioned in Section 1.

2.2. Model-based CF

Also on the basis of the ratings, the main objective of model-based CF is to model the behaviors by machine learning techniques. Through learning from the users' rating logs, the users' preferences hidden in the rating behaviors are thereupon implied. SVM (Support Vector Machine), Decision Tree and Bayesian are the most popular solutions to recognize patterns for classification, which were adopted as the rating classifiers by [3,13,28], respectively. Cremonesi et al. [6] compared the SVD (Singular Value Decomposition)-based algorithms with a non-personalized algorithm based on item popularity. In [10,18,31], matrix factorization was utilized to model characters of items and users by vectors of factors learned from rating patterns. Koren et al. [9] combined the latent factor and neighborhood models to improve the recommendation performance. Zheng et al. [30] proposed a Neighborhood-Integrated Matrix Factorization (NIMF) approach for predictions of collaborative- and personalized-Web service QoS (Quality-of-Service) values. However, the effectiveness of model-based CFs is limited in the rating space that incurs the rating diversity.

2.3. Content-based CF

It is because recommender systems based on users' ratings might encounter problems of rating diversity and rating sparsity that some previous works concentrated their attention on how to take advantage of additional content information such as low-level audio features [23], context information [24], seller information [4], profiles [29], playcounts [11], tags [19,22], etc. to enhance the recommendation. In this paper, we call the ones using tag and playcount information as social-based recommender system. By viewing user, item and tag as three dimensions, Tso-Sutter et al. [25] reduced the three-dimensional correlations to three two-dimensional correlations and then applied a fusion method to re-associate these correlations. Shepitsen et al. [20] proposed a personalized algorithm to induce the users' preferences on music by employing user profiles and tag clusters. Peng et al. [15] proposed a joint item-to-tag recommendation framework utilizing complete information in the tagging data to achieve the recommendation. Qi et al. [16] made attempts to describe users by the inferred user-to-tag ratings so as to improve user-based CF. Konstas et al. [11] adopted playcounts as implicit preferences and performed Random Walks with Restarts (RWR) to predict the preferences by considering users' tagging behaviors. In [11], the experimental results show that, social networking and social tagging play important roles in collaborative recommendation. Although this type of CF has employed additional content information to enhance the recommendation, there is still room for improvement.

3. Proposed approach

3.1. Contribution

To aim at problems existing in rating-based and social-based recommender systems, in this paper, we propose a novel multi-modal recommender system that unifies social and collaborative information to predict users' preferences. Overall the contributions can be summarized as follows.

I. Because social-based recommender systems lack rating information, some recent studies reveal the users' preferences by the user-to-item relevance. However, the user-to-item relevance is not robust enough to reveal the users' preferences. For this issue, in this paper, we propose a formulation to transform playcounts into ratings by statistical calculations. That is, through proposed transformation, the users' implicit votes can be transformed into explicit ones. This provides social-based recommender systems with a solution in predicting users' preferences from rating point of view.

II. Because the rating diversity problem is caused by inconsistent ratings and the rating sparsity problem is caused by insufficient ratings, in this paper, tag information is employed as the complementary information to alleviate such problems. The experimental results show that, without tag information, the high quality of music recommendation is hard to achieve.

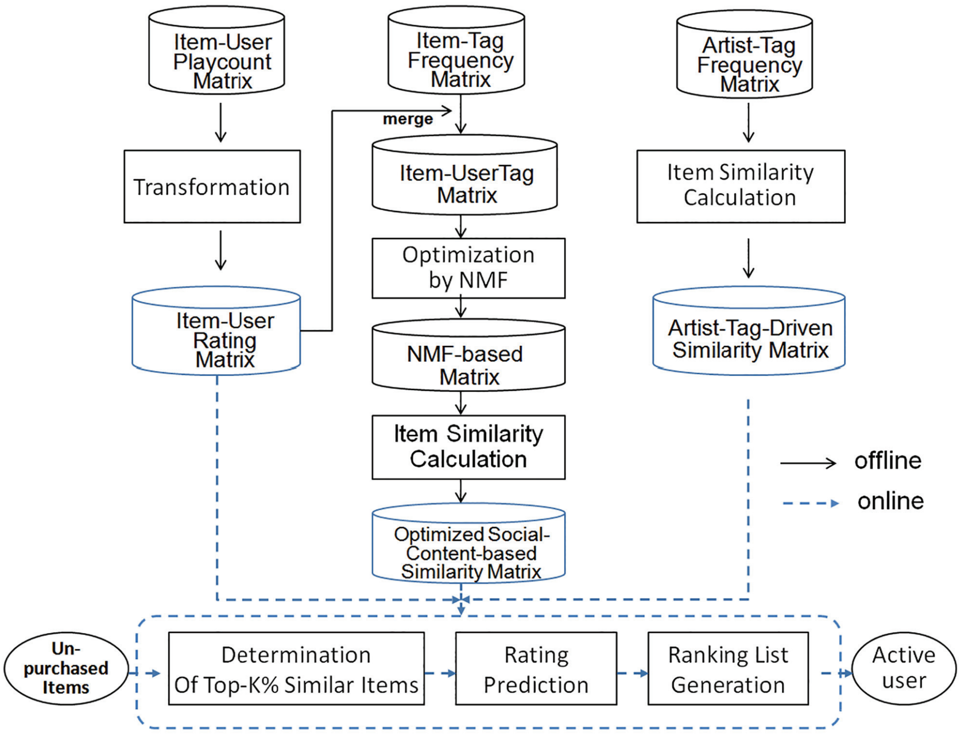

Fig. 1. Framework of proposed approach.

III. To make the performance better, the hybridized information is further optimized by Nonnegative Matrix Factorization (NMF) algorithm. Through the optimization, the latent factors can be discovered and the dimensionality is thereby reduced to increase the prediction precision and decrease the computation cost, respectively. The experimental results show that, the optimized information can really provide neighborhood-based rating prediction with more robust item-similarity.

3.2. Overview of proposed approach

The major difference between our proposed approach and other contemporary approaches is that, we contribute solutions, including rating transformation, integration of social-and-collaborative information and information optimization, to make the preference prediction more effective and efficient. As depicted in Fig. 1, the proposed approach can be decomposed into two stages, namely offline preprocessing and online prediction.

I. Offline preprocessing:
The main purposes of this stage are to meet the requirement of representing the users' preferences by ratings and to accelerate the online prediction. For the requirement of representing the users' preferences by ratings, the item-user playcount matrix is transformed into the item-user rating matrix by statistical calculations. For the acceleration of online prediction, first, the transformed item-user rating matrix and item-playcount matrix are merged into a hybrid matrix. Next, the hybrid matrix is optimized as a NMF-based matrix by performing NMF. Finally, with NMF-based matrix and Artist-Tag frequency matrix, two similarity matrices including an Optimized Social-Content-

based similarity matrix and an Artist-Tag-Driven similarity matrix are generated by calculating similarities among items.

II. Prediction:
Basically, the input of this stage contains un-purchased items, an Optimized Social-Content-based similarity matrix, an Artist-Tag-Driven similarity matrix and an Item-User rating matrix, while the output is a ranking list. The whole procedure starts with the active user's visit. Next, each un-purchased item of the active user is viewed as a target item. For each target item, the k percent most-relevant items are determined by using the Optimized Social-Content-based and Artist-Tag-Driven similarity matrices. According to the relevant items, the rating of each un-purchased item is predicted then. Finally, un-purchased items are sorted by predicted ratings and thereby a ranking list is generated.

3.3. Offline preprocessing

In this stage, the major operations are: 1) rating transformation and 2) constructions of Optimized Social-Content-based and Artist-Tag-Driven similarity matrices. They will be described in the following subsections in detail.

3.3.1. Rating transformation

Generally, for a music recommender system, users' preferences can be represented by two types, namely explicit votes and implicit votes. From usability point of view, the implicit vote is more convenient for users than the explicit vote because the user does not need to give a rating as the explicit vote in a non-rating-based recommender system. As a result, for a social-based recommender system, how to describe the users' preferences using implicit votes is an important issue. As mentioned in [11], user playcount can be viewed as an implicit vote. According to this viewpoint, in this paper, we propose a formulator that projects playcounts onto the rating space to meet the requirement of representing the users' preferences by ratings. Figure 2 shows the scenario of transforming playcounts into ratings. The whole procedure consists of three steps. First, for each user, the playcounts are divided into two ranges by a threshold T, which is defined as:

$$T = \mu - \tau * \sigma, \tag{1}$$

where μ indicates mean of playcount, σ indicates standard deviation of playcounts and τ is a weight parameter. The idea behind Eq. (1) is referred to the traditional votes where the larger the playcount, the higher the user interest. In detail, for a user, the mean of her/his playcounts delivers her/his mean preference. Hence, the playcounts lower than the standard deviation of playcounts are regarded as the presentations of negative preferences, while those higher than the standard deviation of playcounts are regarded as the presentations of positive preferences.

Therefore, in the second step, the range lower than T is further divided into two equivalent sub-ranges with respect to the range number set $\{1, 2\}$, while that higher than T is divided into three equivalent sub-ranges with respect to the range number set $\{3, 4, 5\}$. Third, if a playcount is in the specific range, it can be transformed into the referred range number. In this process, the determination of τ, indeed, is based on the real rating data gathered from a rating system we conducted ever. By referring to the real data distribution, τ is set as 0.5 for the experimental data in this paper. Finally, the item-user rating matrix is constructed as one of input sources for online prediction stage.

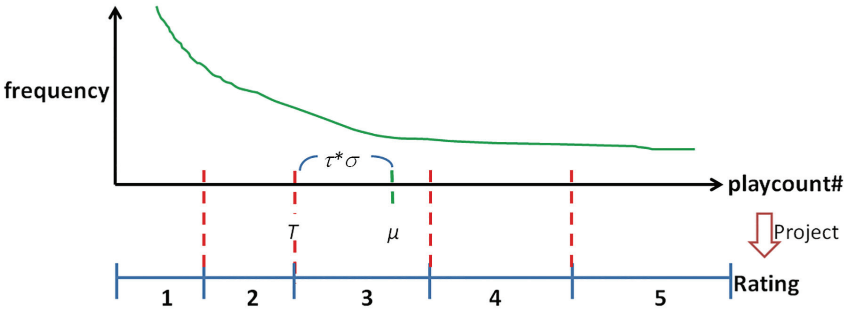

Fig. 2. Scenario of transforming play counts into ratings.

	tag_1	tag_2	tag_3	tag_4
itm_1	1	1	0	0
itm_2	0	0	0.2	0
itm_3	0	1	1	0.6
itm_4	0	0.6	0.3	1

(a)

	u_1	u_2	u_3	u_4
itm_1	1	0.8	0	0.6
itm_2	1	1	0.2	1
itm_3	1	0	0.6	0
itm_4	1	1	0	0

(b)

	u_1	u_2	u_3	u_4	tag_1	tag_2	tag_3	tag_4
itm_1	1	0.8	0	0.6	1	1	0	0
itm_2	1	1	0.2	1	0	0	0.2	0
itm_3	1	0	0.6	0	0	1	1	0.6
itm_4	1	1	0	0	0	0.6	0.3	1

(c)

Fig. 3. Examples of (a): item-tag frequency matrix, (b): item-user rating matrix and (c): Item-UserTag matrix.

3.3.2. Construction of Optimized Social-Content-based similarity matrix

To alleviate problems already described above, we propose an Optimized Social-Content-based similarity matrix constructed by integrating playcount and tag information. Our intent for the integration is to use tag information as a complement to rating (playcount) information. Accordingly, the first operation of this sub-stage is to combine information of playcounts and tags. In this operation, tag information is represented by an item-tag frequency matrix, while playcount information is represented by an item-user playcount matrix. After transforming the item-user playcount matrix into the item-user rating matrix, the item-tag frequency matrix and item-user rating matrix are merged into a hybrid information matrix, namely Item-UserTag matrix which can be defined in Definition 1. Note that, the element values in both of the item-tag frequency and the item-user rating matrixes are the normalized item-tag frequencies and the normalized item-user ratings, respectively.

Definition 1. Assume that, there are a number of unique users $U = \{u_1, u_2, \ldots, u_s, \ldots, u_{|U|}\}$, a number of unique items $I = \{itm_1, itm_2, \ldots, itm_i, \ldots, itm_{|I|}\}$ and a number of unique tags $T = \{tag_1, tag_2, \ldots, tag_j, \ldots, tag_{|T|}\}$ in the database. Given an item-user rating matrix $IUR_{I\to U}$ and an item-tag frequency matrix $ITF_{I\to T}$, the Item-UserTag matrix is defined as:

$$IUT_{I\to UT}[v_{i,m}],$$

Table 2
Example of Optimized Social-Content-based similarity matrix

	itm_1	itm_2	itm_3	itm_4
itm_1	1	0.98	0.197	0.69
itm_2		1	0	0.533
itm_3			1	0.846
itm_4				1

where UT is the set of combining sets U and T, v is the normalized value ranged from 0 to 1, $|UT| = |U|+|T|$ and $0 < m \leqq |UT|$. For example, Figs 3-(a), (b) and (c) show examples of an item-tag frequency matrix, an item-user rating matrix and an Item-UserTag matrix, respectively.

After generating the Item-UserTag matrix, the second operation of this sub-stage is to perform NMF algorithm to approximate a better information matrix called NMF-based matrix. Actually, NMF is a popular matrix factorization method [5,12,14,27] aiming at the factor analysis of matrices whose elements are nonnegative. As a result, our intents for this operation are to reduce the dimensionality and to discover the latent factors to make the prediction more efficient and effective, respectively. Based on the Definition 1, in this operation, the Item-UserTag matrix is approximated as a product of two sub-matrices:

$$IUT_{I \to UT}[v_{i,\, m}] \;\approx\; IF_{I \to F}[iv_{i,\, f}] \cdot UTF_{UT \to F}[utv_{m,\, f}]^T$$

where *IF* and *UTF* indicate the factor matrices, each item i and user m are modeled by a factor vector set F, $0 < f \leqq |F|$, and elements of two matrices are all nonnegative. In this process, the objective function,

$$\left\| IUT - IF \cdot UTF^T \right\|^2 = \sum_{i,\, m} (v - \sum_{f=1}^{|F|} iv_{i,\, f} - utv_{m,\, f})^2, \tag{2}$$

can be minimized through two iterative update algorithms [5,12,14,27],

$$iv_{i,\, f} \leftarrow iv_{i,\, f} \frac{(IUT \cdot UTF)_{i,\, f}}{(IF \cdot UTF^T \cdot UTF)_{i,\, f}} \text{ and } utv_{m,\, f} \leftarrow utv_{m,\, f} \frac{(IUT^T \cdot IF)_{m,\, f}}{(UTF \cdot IF^T \cdot IF)_{m,\, f}}.$$

According to the NMF method, the *IUT* is decomposed into two factor matrices including *IF* and *UTF*, and *IF* called NMF-based matrix in this paper is finally utilized for the next operation. In this work, $|F|$ is set as 1000. Based on the example of Fig. 3, the examples of factor matrixes are shown in Fig. 4.

On the basis of the NMF-based matrix, the final operation of this sub-stage is to yield the Optimized Social Content-based similarity matrix by calculating similarities among items. Behind the Optimized Social Content-based similarity matrix, the primary purpose is to reduce the online prediction cost. The item similarity is defined in Definition 2.

Definition 2. Following Definitions above, there are $|I|$ unique items and $|F|$ dimensions in the NMF-based matrix. The feature vectors of items x and y are represented as $\{iv_{x,1}, iv_{x,2}, \ldots, iv_{x,|F|}\}$ and $\{iv_{y,1}, iv_{y,2}, \ldots, iv_{y,|F|}\}$, respectively. Accordingly, the Optimized Social-Content-based similarity between itm_x and itm_y is defined as:

$$OSCsim(itm_x,\ itm_y) = \frac{\sum_{0<q\leqslant|F|} iv_{x,\, q} * iv_{y,\, q}}{\sqrt{\sum_{0<q\leqslant|F|} (iv_{x,\, q})^2} * \sqrt{\sum_{0<q\leqslant|F|} (iv_{y,\, q})^2}}. \tag{3}$$

	f_1	f_2
itm_1	**1.6545**	**0.3321**
itm_2	**1.6057**	**0**
itm_3	**0**	**1.9555**
itm_4	**0.7382**	**1.1705**

(a)

	u_1	u_2	u_3	u_4	tag_1	tag_2	tag_3	tag_4
f_1	**0.5514**	**0.5924**	**0**	**0.4611**	**0.2819**	**0.2294**	**0**	**0.0165**
f_2	**0.5164**	**0.1146**	**0.2264**	**0**	**0**	**0.5153**	**0.4499**	**0.4484**

(b)

Fig. 4. Examples of (a): *IF* matrix, (b): *UTF* matrix.

Based on Definition 2, the similarities among items can be calculated as elements of the Optimized Social-Content-based similarity matrix. The similarity matrix is defined in Definition 3.

Definition 3. According to Definitions 1 and 2, there are $|I|$ unique items. The Optimized Social-Content-based similarity matrix is defined as:

$$OSC_{I \to I}[OSCsim_{i,i}],$$

where *OSCsim* denotes the Optimized Social Content-based similarity between two items. Table 2 is the example of Optimized Social-Content-based similarity matrix by following above examples.

3.3.3. Construction of artist-tag-driven similarity matrix

In addition to information of item tags, another valuable one that can enhance the music recommendation is information of artists. It is motivated by the observation that, most users' musical preferences are represented well stably for some specific artists. That is, it is effective to mine the users' musical preferences from the preferable artists. This point inspires us to take advantage of artist tags as useful information to imply the users' preferences. In this sub-stage, the first operation is to generate the artist-tag frequency matrix, which can be defined in Definition 4.

Definition 4. Based on Definition 1, assume there are n unique artists $AT = \{art_1, art_2, \ldots, art_a, \ldots, art_n\}$ in the database and an artist art_a contains a set of performed items where $art_a = \cup itm_z$, $itm_z \in I$. Given the item-tag frequency $ITF_{I \to T}[itf_{i,j}]$, the artist-tag frequency matrix is defined as:

$$ATX_{AT \to T}[atf_{a,j}],$$

where

$$atf_{a,\,j} = \frac{\sum_{itm_z \in art_a} itf_{itm_z,\,j}}{|art_a|}.$$

Figure 5 is an example to illustrate how to generate the artist-tag frequency matrix. Following the example in Fig. 3, suppose there are two unique artists and four unique tags in the database, where

Table 3
Example of artist-tag-driven similarity matrix

	itm_1	itm_2	itm_3	itm_4
itm_1	1	1	0.815	0.815
itm_2		1	0.815	0.815
itm_3			1	1
itm_4				1

		tag_1	tag_2	tag_3	tag_4
art_1	itm_1	1	0.8	0	0.6
	itm_2	1	1	0.2	1
art_2	itm_3	1	0	0.6	0
	itm_4	1	1	0	0

(a)

	tag_1	tag_2	tag_3	tag_4
art_1	1	0.9	0.1	0.8
art_2	1	0.5	0.3	0

(b)

Fig. 5. Example of generating the artist-tag frequency matrix; (a): original item-tag frequency matrix; (b): transformed artist-tag frequency matrix.

$art_1 = \{itm_1, itm_2\}$ and $art_2 = \{itm_3, itm_4\}$. Based on Definition 4, the tag feature vectors for art_1 and art_2 are $\{(1 + 1)/2, (0.8 + 1)/2, (0 + 0.2)/2, (0.6 + 1)/2\} = \{1, 0.9, 0.1, 0.8\}$ and $\{1, 0.5, 0.3, 0\}$, respectively.

After the artist-tag frequency matrix is generated, the artist-tag-driven similarity matrix can be derived by calculating similarities among items, which is defined in Definition 5.

Definition 5. Assume that, itm_x and itm_y are contained in art_a and art_b, respectively. Based on the artist-tag frequency matrix, the item similarity between itm_x and itm_y using artist-tag-driven similarity is defined as:

$$ATsim(itm_x,\ itm_y) = ATsim(art_a, art_b) = \frac{\sum_{0<p\leqslant|T|} atf_{a,\,p} * atf_{b,\,p}}{\sqrt{\sum_{0<p\leqslant|T|} (atf_{a,\,p})^2} * \sqrt{\sum_{0<p\leqslant|T|} (atf_{b,\,p})^2}}. \quad (4)$$

The concept behind Definition 5 is that the item similarity can be represented by the artist similarity. That is, if two users prefer the same artists, they might have the same preferences on music. Consequently, the artist-tag-driven similarity matrix is derived by computing the similarities among items, which can be defined in Definition 6.

Definition 6. Based on Definitions 1, 4 and 5, assume there are n unique artists $AT = \{art_1, art_2, \ldots, art_a, \ldots, art_n\}$ in the database. The artist-tag-driven similarity matrix is defined as:

$$ATS_{I\to I}[ATsim_{i,i}],$$

where $ATsim$ denotes the artist-tag-driven similarity between two items. Table 3 is the example of the artist-tag-driven similarity matrix based on Fig. 5 and Definition 6.

Table 4
Example of the fusion similarity matrix

	itm_1	itm_2	itm_3	itm_4
itm_1	1	0.98	0.747	0.562
itm_2		1	0	0.434
itm_3			1	0.846
itm_4				1

3.3.4. Construction of fusion similarity matrix

On the basis of Optimized Social-Content-based and Artist-Tag-Driven similarities, we fuse these two similarities as a fusion similarity shown in Definition 7.

Definition 7. Assume that, itm_x and itm_y are contained in art_a and art_b, respectively. Given that, the related social-content-based and artist-tag-driven similarities are $OSCsim(itm_x, itm_y)$ and $ATsim(itm_x, itm_y)$, respectively. The fusion similarity between itm_x and itm_y can be defined as:

$$FUsim(itm_x, itm_y) = OSCsim(itm_x, itm_y) * ATsim(itm_x, itm_y). \tag{5}$$

By considering the Tables 2 and 3, Table 4 is the example of the fusion similarity matrix.

3.4. Online prediction

Because the item similarities have been generated in the offline preprocessing stage, the main goal of online prediction is to infer ratings of un-purchased items (called unknown ratings in this paper) by employing the generated item similarities. The whole procedure of online prediction is triggered by an active user visit. Then the unknown ratings are predicted one by one. After all unknown ratings are predicted, the mechanism generates a ranking list by sorting the un-purchased items. In detail, the prediction firstly determines top k percent items relevant to the target item by the item similarities. Next, for an unknown rating, it can be calculated by Definition 8.

Definition 8. Based on Definitions above, given a user-item rating matrix $UIR_{U\to I}[r_{s,i}]$ which is the transpose of item-user rating matrix mentioned in Section 3.3.1. Assume the most relevant item set to a target item itm_x for an active user u_s is $RI_s = \cup itm_c$, where $itm_x \in I$ and $itm_c \in I$. Thereby the rating of the un-purchased itm_x for u_s is defined as:

$$\hat{r}_{s,\,x} = \frac{\sum_{itm_c \in RI_s} sim_{x,\,c} * r_{s,\,c}}{\sum_{itm_c \in RI_s} sim_{x,\,c}} \tag{6}$$

where

$$\begin{cases} sim_{x,\,c} = OSCsim(itm_x, itm_c), \text{ if using optimized social-content-based similarity} \\ sim_{x,\,c} = ATsim(itm_x, itm_c), \text{ if using artist-tag-driven similarity} \\ sim_{x,\,c} = FUsim(itm_x, itm_c), \text{ if using fusion similarity} \end{cases}$$

Note that, according to three similarity matrices, the prediction models can be defined as:

$$\begin{cases} \textit{Optimized Social Content-based Prediction (OSCP) for optimized social-content-based similarity} \\ \textit{Artist-Tag-Driven Prediction (ATP) for artist-tag-driven similarity} \\ \textit{Multi-modual Music Recommender system (MMR) for fusion similarity} \end{cases}$$

Table 5
Descriptions of the experimental data

Description	Dataset 1	Dataset 2
#Users	912	319
#Items	27,303	40992
#Tags	3,937	235921
#Artists	3,559	8553
#User-Item rating	204,164	64757
#Item-Tag frequency	261,986	1718257

Following examples above, assume a user-to-item rating matrix is shown as Table 1. If the target item for an active user u_3 is *itm*$_1$, the target rating $\hat{r}_{3,\ 1}$ is:

$$\begin{cases} \frac{0.98*1+0.197*0+0.69*5}{0.98+0+0.69} = 2.65, \textit{if using optimized social-content-based similarity} \\ \frac{1*1+0.815*0+0.815*5}{1+0+0.815} = 2.79, \textit{if using artist-tag-driven similarity} \\ \frac{0.98*1+0.747*0+0.562*5}{0.98+0+0.562} = 2.47, \textit{if using fusion similarity.} \end{cases}$$

4. Empirical study

In the preceding section, our proposed approach has been presented in great detail. In this section, we will show the results of evaluating our proposed approach through complete experiments.

4.1. Experimental data

The experimental data came from the collection of user listening behaviors provided by Last.fm [1]. Last.fm is a popular social music website that provides users with online listening and tagging services. Through this platform, 30 million active users can describe their music tastes by tagging the music they have listened to. Therefore, the data from Last.fm is widely employed as the experimental data. Our experimental data contains two sets, namely Datasets 1 and 2. As shown in Table 5, the Dataset 1 consists of 912 users, 27303 items and 3937 tags, while the Dataset 2 is composed of 319 users, 40992 items and 235921 tags. The major difference is that, the Dataset 1 contains the logs in 2010 and the dataset contains the logs in 2013. In our experiments, for each user, around 20% of rated items were randomly selected as the testing data, and the other rated items were used as the training data. Table 5 gives a detailed description of the experimental data. Note that, the dimensionality of the optimized matrix, *IF*, is reduced to 1000 in this work. For the rating transformation mentioned in Section 3.3.1, τ is set as 0.5.

4.2. Evaluation measures

As we can recall from Section 1, the recommendation procedure can be decomposed into two stages, namely rating prediction and item selection. Two types of measures for these two stages were utilized in previous works, namely *rating error* and *precision*. In general, *rating error* adopted as the evaluation measure for the rating prediction stage indicates the difference between the predicted rating and ground truth, while *precision* adopted as the evaluation measure for the item selection stage indicates proportion of the testing items against the top N results on the ranking list (Top-N Recommendation, TNR).

Table 6
Example of predicted ranking results

e	Item	*predicted rating*	rel_e
1	itm_4	4.7	0.9
2	itm_6	4.2	NULL
3	itm_1	3.8	0.4
4	itm_3	3.5	0.7
5	itm_5	3.3	0.1
6	itm_2	3.0	0.2

However, it is not easy to evaluate TNR-based recommender systems using traditional *precision* since the ranking list contains both un-purchased and testing (rated) items. For traditional *precision*, TNR-based recommender systems view testing items as the ground truth. Therefore, for this type of recommender systems, a successful prediction lies in an aspect that, a resulting item should be one of testing items. That is, the items that are not testing items are regarded as incorrect results. This measurement paradigm seems unsuitable since there is no evidence that un-purchased items are negative for an active user. A simple example is used to explain this point. Assume there are 6 items {item1, item2, item3, item4, item5, item6} in the database. For an active user, the testing item set, which indicates the items rated by the active user, is {item1, item2}. Then, suppose the ranking list derived by a TNR-based recommender system is {item3, item2, item4, item1, item6, item5}. For a TNR-based recommender system, the precision is defined as (#*correct*/N) where #*correct* denotes the number of positive items and N is the number of resulting items. Accordingly, if N value is 2, the resulting items in this example are {item3, item2}. Obviously, only item2 is correct and the precision is therefore 1/2. On the whole, the precisions in this example are 0/1, 1/2, 1/3, 2/4, 2/5 and 2/6 where the N values are 1, 2, 3, 4, 5 and 6, respectively. This example shows that, it is not feasible to identify the un-purchased items as false predictions since item3, item4 and item5 may be the positive items for the active user.

Therefore, in this paper, a robust measure called NDCG (Normalized Discount Cumulative Gain) is used as the evaluation metric. In this evaluation, for each user, all testing items are firstly sorted by normalized playcounts, where playcounts are regarded as relevance judgments or scores. Note that, testing items in this evaluation are viewed as unrated items so the unrated items consist of testing items and un-purchased items. After ratings of all unrated items are predicted, a ranking list consisting of testing items and un-purchased items is generated. The top N testing items are then selected from the ranking list. The NDCG is defined as:

$$NDCG = \frac{1}{IDCG} * \sum_{e=1}^{N} \frac{2^{rel_e} - 1}{\log_2(e-1)}, \tag{7}$$

where IDCG (Ideal Normalized Discount Cumulative Gain) is a normalization factor, e indicates the ranking positions and *rel* indicates the relevance judgment or score using normalized playcounts. From above definition of NDCG, the major difference between traditional *precision* and NDCG is that, the evaluation using NDCG considers testing items only, but the evaluation using traditional *precision* considers both testing items and un-purchased items.

Tables 6 and 7 are examples to illustrate how to calculate NDCG. Assume that, there are 6 unrated items in this evaluation, which contains a testing item set $\{itm_1, itm_2, itm_3, itm_4, itm_5\}$ and an un-purchased item set $\{itm_6\}$. In Table 6, the ranking set is $\{itm_4, itm_6, itm_1, itm_3, itm_5, itm_2\}$, where the predicted rating set is {4.7, 4.2, 3.8, 3.5, 3.3, 3.0} and the related relevance score set is {0.9, NULL, 0.4, 0.7, 0.1, 0.2}. In this example, because no relevance score is assigned to itm_6, it cannot be the

Table 7
Example of ideal ranking results

e	Item	rel_e
1	itm_4	0.9
2	itm_3	0.7
3	itm_1	0.4
4	itm_2	0.2
5	itm_5	0.1

consideration for the evaluation. Accordingly, the DCG is 1.468 without considering the un-purchased item itm_6. Similarly, the IDCG which is 1.511 can be derived by Table 4. Thus the NDCG is 1.468/1.511 = 0.97.

In addition to NDCG, the other metric used for evaluations is RMSE (Root Mean Square Error), which is defined as:

$$RMSE = \sqrt{\frac{\sum_{|test|} (r - \hat{r})^2}{|test|}}, \tag{8}$$

where r stands for the ground truth, $\hat{r}$ stands for the predicted rating value and *test* stands for the testing dataset. On the whole, NDCG shows the ratio of predicted ranking against ideal ranking, while RMSE shows the error variance between predicted ratings and ground truths. That is, the lower the RMSE, the lower the error, the higher the precision, the better the recommendation. In contrast, the higher the NDCG, the better the recommendation. To ensure robust experimental evaluations, both NDCG and RMSE are used to verify the proposed method in the experiments.

4.3. Experimental results

In the experiments, the evaluations were conducted in two main concepts: 1) Effectiveness evaluations of our proposed individual models, including OSCP, ATP and MMR, 2) Effectiveness comparisons between our proposed fusion model and existing well-known recommender systems in terms of RMSE and NDCG, and 3) Efficiency evaluations of all compared methods.

4.3.1. Effectiveness of evaluating our proposed individual models on Dataset 1

Before evaluating our proposed multi-modal recommender system, what we have to clarify is the effectiveness of proposed individual models. In this evaluation, top k percent relevant items referred to Section 3.4 are adopted as the basis of prediction. Figure 6 is the experimental result that shows some important points. First, the RMSE increases as the relevant items increase. This delivers an aspect that, as the relevant items increase, the noises increase so that the performance decreases. Second, OSCP performs better than ATP, and MMR is the best. From this result, we can obtain that, without OSCP or ATP, the fusion model cannot achieve high quality of music recommendation. In summary, OSCP and ATP, indeed, play crucial roles in the fusion model. Since MMR is the best prediction model, we adopted it as the main proposed method to compare with other well-known recommender systems.

4.3.2. Effectiveness comparisons between our proposed fusion model and existing well-known recommender systems

4.3.2.1. Compared methods

In order to make the experiments solid, we selected 15 state-of-the-art recommender systems as compared ones, including memory-based, content-based and model-based recommender systems. Table 8

Table 8
List of compared methods

Method	Category
User-Based (UB) [17]	Memory-Based
Item-Based (IB) [21]	Memory-Based
Similarity Fusion (SF) [26]	Memory-Based
Significance-Based (SB) [2]	Memory-Based
User-based Prediction by Tag Ratings (UPTR) [16]	Content-Based
RandomWalk (RWR) [11]	Content-Based
Tag-aware recommender system (TagA) [25]	Content-Based
SVM [28]	Model-Based
Decision Tree (DT) [13]	Model-Based
Bayes [3]	Model-Based
PureSVD [6]	Model-Based
SVD++ [9]	Model-Based
Neighborhood Integrated Matrix Factorization (NIMF) [30]	Model-Based
Matrix Factorization (MF) [10]	Model-Based
Nonnegative Matrix Factorization (NMF) [31]	Model-Based

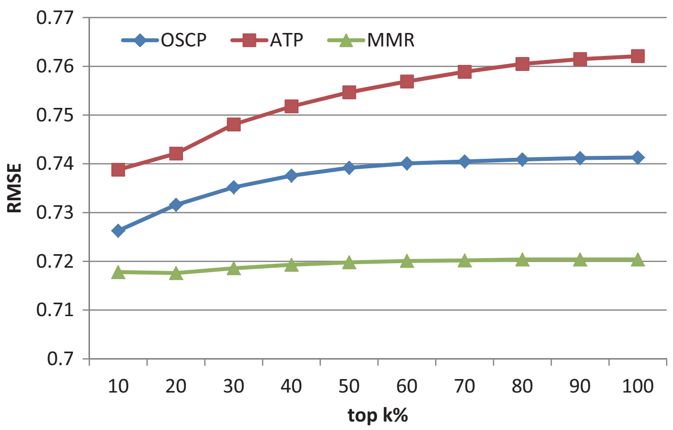

Fig. 6. Experimental results of the proposed OSCP, ATP and MMR in terms of RMSE.

depicts the information of compared approaches in the experiments. Most of the listed recommender systems are well-known ones, which have been described in Section 2 briefly. Note that, since some compared methods generate the ranking list only, this type of methods is evaluated by NDCG.

4.3.2.2. Experimental analysis for issue of rating sparsity on Dataset 1

As the first step in our experimental analysis, we will examine our proposed method MMR for dealing with problem of rating sparsity. As shown in Table 9, in this experiment, we totally selected 40866 item ratings as unknown values to predict. However, because traditional IB encounters problem of rating sparsity, only 34870 item ratings can be predicted successfully by IB. That is, there are 5996 item ratings which cannot be predicted by IB. In contrast, there are just 66 item ratings which cannot be predicted by our proposed MMR. Clearly, the coverage of rating predictions of our proposed MMR is much higher than that of traditional IB. It reveals that, our proposed method can really alleviate problem of rating sparsity significantly. Based on this analysis, the testing item ratings are set as 34870 for all compared methods in the following experiments.

Table 9
Comparisons of our proposed method and traditional IB in terms of coverage

	#item ratings selected	#item ratings predicted (#testing item ratings)	#item ratings un-predicted
IB	40866	34870	5996
MMR	40866	40800	66

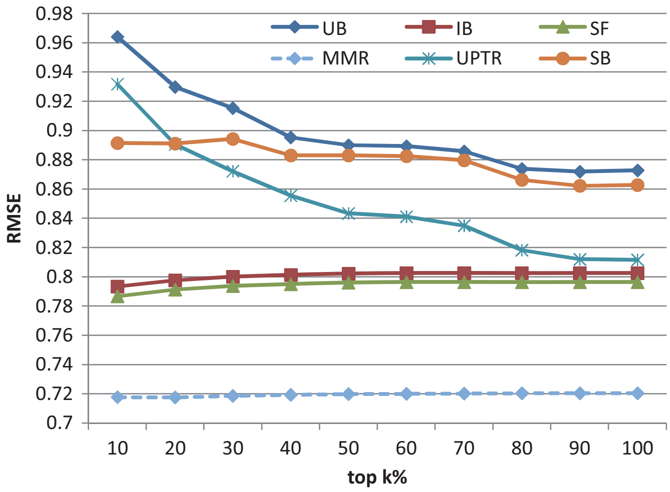

Fig. 7. Experimental results of comparing MMR and existing recommender systems in terms of RMSE.

4.3.2.3. Evaluations using RMSE on Dataset 1

Since rating sparsity and rating diversity might bring out unsatisfactory prediction results, they need to be investigated by detailed experimental evaluations. Hence, in the following experiments, the compared methods are evaluated using RMSE and NDCG to show the abilities of coping with problems of rating sparsity and rating diversity. First, the RMSE comparison of different methods is shown in this sub-Section. Note that, top k percent relevant items/users are adopted as the basis of prediction in this experiment.

Figure 7 depicting the experimental results for RMSE delivers some important aspects. First, the best one is MMR, while UB is the worst. Second, user-based methods are worse than item-based ones. Third, the results of item-based and similarity-fusion (SF) methods are pretty close. Fourth, although UPTR is a content-based method, it is not better than item-based and SF methods. The potential explanation is that, the recommendation result of UPTR is generated based on user-based idea. Fifth, RMSEs of methods UB and UPTR decrease as the relevant users increase. This result is very different from those of IB, SF and MMR methods. It delivers an important aspect that, more users bring out better results for user-based methods, but more items cannot facilitate item-based predictions. The potential interpretation for this result is that, the users are very relevant on the similar listening and tagging behaviors, but items are not.

It is because model-based SVM, DT, Bayes, SVD++, MF, NMF and NIMF predict the ratings without considering top k percent relevant items that we conducted an overall evaluation of comparing the best RMSEs of model-based methods, memory-based methods and our proposed MMR. From Fig. 8, we can obtain that, first, Bayes is the worst. Second, the performances of SVD++, IB, MF, NMF and SF are pretty close and slightly better than that of UPTR. Third, overall item-based and SF methods outperform

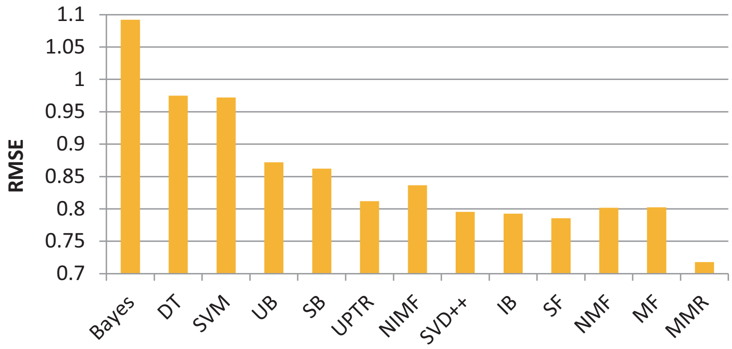

Fig. 8. Overall experimental results of comparing MMR and existing recommender systems in terms of RMSE.

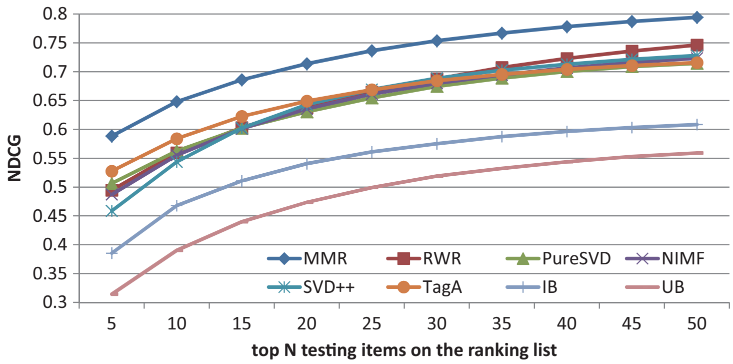

Fig. 9. Experimental results of comparing MMR and existing recommender systems in terms of NDCG.

user-based and model-based ones. Fourth, although SF fuses user-based and item-based CFs, it still cannot bring out significantly better results than item-based CF. On the whole, the experimental results show that our proposed method is better than other contemporary ones in terms of RMSE. It says that, from RMSE point of view, our proposed method can predict the users' ratings effectively by fusing the optimized social-content-based and artist-tag-driven similarities.

4.3.2.4. Evaluations using NDCG on Dataset 1

The final evaluation is to examine the methods using NDCG, including comparisons of MMR, RWR, PureSVD, NIMF, SVD++, IB, TagA and UB. Figure 9 depicts the experimental results in terms of NDCG, and some related observations are described as follows. First, UB is the worst and MMR is the best. The NDCG of MMR can reach around 0.8 while considering 10 percent relevant items in this experiment. Second, the results of content-based and model-based methods are very close and better than those of memory-based ones. It shows that, results using RMSE cannot reveal the real strength of recommender systems in predicting the users' preferences. In overall, our proposed method can make the more promising results than compared methods for music recommendation.

4.3.2.5. Scalability evaluations using RMSE on Dataset 1

The goal of this experiment is to clarify the effectiveness under different sizes of training sets. Figure 10 shows the experimental results that, the more the training data, the higher the effectiveness. However, the effectiveness between the training sizes of 80% and 90% is pretty close and that of 70% and 80% is slightly different. It further says that, the proposed method performs stably on different training sizes.

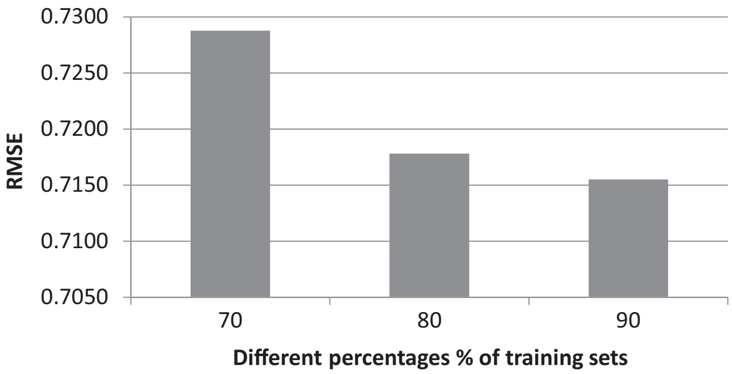

Fig. 10. Experimental results for scalability evaluations using RMSE on Dataset 1.

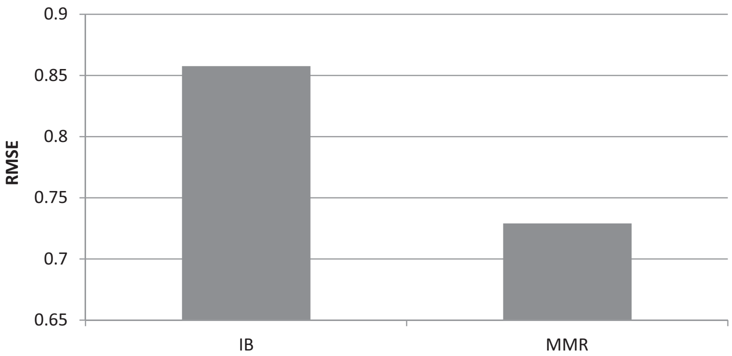

Fig. 11. Experimental results of comparing MMR with IB in terms of RMSE on Dataset 2.

4.3.2.6. Evaluations using RMSE and NDCG on Dataset 2

Through the above evaluations on Dataset 1, we can know that, our proposed method is better than other state-of-the-arts methods on music recommendation in terms of RMSE, NDCG and sparsity. To make the evaluations more robust, we conducted more evaluations using Dataset 2. As shown in Table 5, there exist different listening and tagging behaviors in these two datasets. Therefore, our intent is to investigate whether the effectiveness changes as the listening and tagging behaviors are different. Because the best compared methods using Dataset 1 are IB and RWR for RMSE and NDCG, respectively, IB and RWR were selected as the compared methods using Dataset 2. Whatever for RMSE or NDCG, Figs 11 and 12 reveal the proposed method MMR still performs better than IB and RWR, respectively even using Dataset 2. In detail, our proposed method can really achieve high quality of music recommendation even the listening and tagging behaviors change.

4.3.3. Efficiency evaluation results

In addition to the effectiveness evaluations, another issue for real applications is the efficiency for each active user. To address this issue, we conducted a comparative experiment that shows the prediction time of the compared methods for a user. Table 10 reveals the experimental results. As a whole, almost the methods can achieve the prediction for a user in a second. Only SVM and RWR need time larger than a second. From usage point of view, it is practical if the results can be returned in around one second. Therefore, only RWR cannot satisfy the need of time. Moreover, although RWR can really achieve high effectiveness, the execution time is so high that the users cannot accept it.

Table 10
Prediction time of compared methods

Method	Prediction time (Sec.)
UB	8.7719E-05
IB	0.0032807
SF	0.00377961
SB	8.7719E-05
UPTR	8.7719E-05
RWR	48.28947368
TagA	0.00338596
SVM	1.5372807
DT	0.02302632
Bayes	0.01927632
PureSVD	0.0685307
SVD++	0.00430921
NIMF	9.3421E-05
MF	6.8421E-05
NMF	5.1316E-05
MMR	0.00307

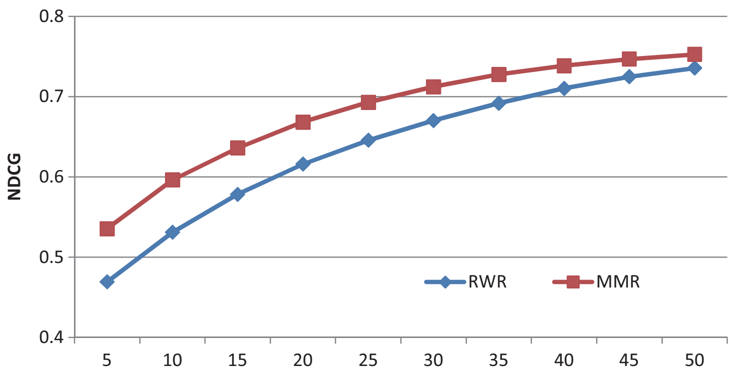

Fig. 12. Experimental results of comparing MMR with RWR in terms of NDCG on Dataset 2.

4.4. *Discussion for experimental results*

Based on the experimental evaluations, we summarize the discovery and discussion as follows:

I. In the experiments, OSCP combines the rating and tag information, while IB just utilizes the rating information. By comparing OSCP with IB using Figs 5, 6 and 7, OSCP performs better than IB. This result reveals that the combined information is more robust than the rating information in dealing with problems of rating diversity and rating sparsity. It also arouses an echo in [11] that tag information does facilitate recommender systems in predicting users' preferences.

II. From Figs 6, 7 and 8, we can obtain that, our proposed method is the best in terms of both RMSE and NDCG. An important discovery from these results is that, it is not easy to perform well stably for both RMSE and NDCG. For example, IB is good for RMSE, but poor for NDCG. According to our observation, although IB can predict the ratings close to the ground truth, the related ranking cannot approximate the truth. In contrast to IB, SVD++ performs well stably in both RMSE and NDCG. Therefore, what we want to address here is *not* that, the methods good for RMSE are poor for NDCG. Instead, the methods which are good for both RMSE and NDCG are really reliable for music recommendation.

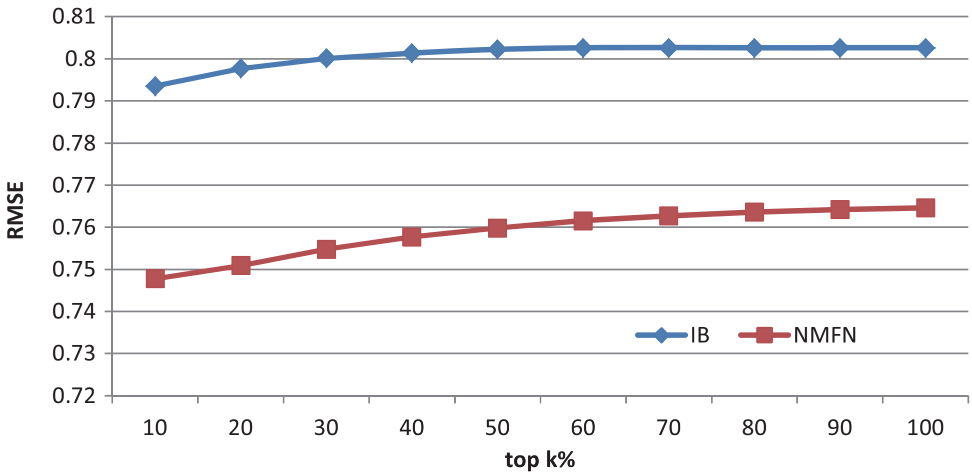

Fig. 13. Experimental results of comparing NMFN and IB in terms of RMSE.

III. Actually, some compared methods in this paper are fusion-based ones that adopt several information sources and combine different CFs into a fusion mechanism such as MMR, SF, NIMF, SVD++, TagA, RWR, UPTR and so on. However, the experimental results reveal that, not all fusion-based methods are good for both RMSE and NDCG. On average, our proposed method can bring out the robust results in terms of RMSE and NDCG.

IV. In addition to above experimental analysis, an idea of using NMF-based optimization needs to be justified here. For this justification, we completed one more method, namely NMF-based Neighborhood prediction (NMFN) using the optimized rating matrix (user-item rating matrix). Figure 13 shows the relative experimental result that, the method NMFN using the optimized rating matrix performs better than IB using the rating matrix without optimization. This experimental result delivers an aspect that the optimization operation can really facilitate the rating prediction. Furthermore, although NMFN has adopted the optimized rating matrix, it cannot still outperform MMR that utilizes the integrated and optimized information.

5. Conclusion and future work

In fact, it is difficult to narrow the gap between users' preferences and music because of problems of rating diversity, rating sparsity and lack of ratings. This paper presents a novel music recommender system named Multi-modal Music Recommender system (MMR) that can bridge users' preferences to music effectively by integrating social and collaborative information.

In summary, the major contributions are described as follows. First, by transforming playcounts into ratings, problem of lack of ratings can be avoided successfully. Second, by integrating rating and tag information, problems of rating diversity and rating sparsity can be alleviated significantly. Third, NMF-based optimization can effectively reduce computation cost and improve the prediction quality. The experimental results reveal that, our proposed method MMR performs better than other 15 state-of-the-art recommender systems in terms of RMSE and NDCG. It delivers the perspective that, our proposed MMR can associate users' preferences with music more successfully than existing recommender systems.

In the future, there are some issues to investigate further. First, the effect of music low-level features will be studied by adding music low-level features into our proposed method. Second, other context information like emotion, environment information and so on will be considered to satisfy increasing needs of ubiquitous recommendation. Third, this idea will be applied to other types of multimedia recommender systems in the future.

Acknowledgement

This research was supported by Ministry of Science and Technology, Taiwan, R.O.C. under grant no. MOST 105-2221-E-230-011-MY2 and MOST 105-2632-S-424-001.

References

[1] <http://www.last.fm>.
[2] J. Bobadilla, A. Hernando, F. Ortega and A. Gutiérrez, Collaborative filtering based on significances, *Information Sciences* **185**(1) (2012), 1–17.
[3] J.S. Breese, D. Heckerman and C. Kadie, Empirical Analysis of Predictive Algorithms for Collaborative Filtering, *Proc. Int. Conf. on Uncertainty in Artificial Intelligence*, 1998, pp. 43–52.
[4] L.S. Chen, F.H. Hsu, M.C. Chen and Y.C. Hsu, Developing recommender systems with the consideration of product profitability for sellers, *Information Sciences* **178**(4) (2008), 1032–1048.
[5] D. Cai, X. He, J. Han and T.S. Huang, Graph Regularized Nonnegative Matrix Factorization for Data Representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(8) (2011), 1548–1560.
[6] P. Cremonesi, Y. Koren and R. Turrin, Performance of recommender algorithms on top-n recommendation tasks, *Proc.of the fourth ACM conference on Recommender systems*, 2010, pp. 39–46.
[7] M. Deshpande and G. Karypis, Item-based top-N recommendation algorithms, *ACM Transactions on Information Systems.* **22**(1) (2004), 143–177.
[8] J.L. Herlocker, J.A. Konstan, A. Borchers and J. Riedl, An algorithmic framework for performing collaborative filtering, *Proc. of the international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 230–237.
[9] Y. Koren, Factorization meets the neighborhood: A multifaceted collaborative filtering model, *Proc. of the ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 426–434.
[10] Y. Koren, R. Bell and C. Volinsky, Matrix Factorization Techniques for Recommender Systems, *Computer* **42**(8) (2009), 30–37.
[11] I. Konstas, V. Stathopoulos and J.M. Jose, On social networks and collaborative recommendation, *Proc. of the ACM SIGIR conference on Research and development in information retrieval*, 2009, pp. 195–202.
[12] D.D. Lee and H.S. Seung, Algorithms for Non-Negative Matrix Factorization, *Advances in Neural Information Processing Systems* **13**, MIT Press, 2001.
[13] D. Nikovski and V. Kulev, Induction of Compact Decision Trees for Personalized Recommendation, *Proc. Int. Conf. on the ACM symposium on Applied computing*, 2006, pp. 575–581.
[14] P. Paatero and U. Tapper, Positive Matrix Factorization: A Non-Negative Factor Model with Optimal Utilization of Error Estimates of Data Values, *Environmetrics* **5**(2) (1994), 111–126.
[15] J. Peng, D.D. Zeng, H. Zhao and F. Wang, Collaborative filtering in social tagging systems based on joint item-tag recommendations, *Proc. of ACM international conference on Information and knowledge management*, 2010, pp. 809–818.
[16] Q. Qi, Z. Chen, J. Liu, C. Hui and Q. Wu, Using inferred tag ratings to improve user-based collaborative filtering, *Proc. of Annual ACM Symposium on Applied Computin*, 2012, pp. 2008–2013.
[17] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom and J. Riedl, GroupLens: An Open Architecture for Collaborative Filtering of Netnews, *Proc. Int. Conf. on ACM Computer Supported Cooperative Work*, 1994, pp. 175–186.
[18] H. Shan and A. Banerjee, Generalized Probabilistic Matrix Factorizations for Collaborative Filtering, *Proc. of IEEE International Conference on Data Mining*, 2010, pp. 1025–1030.
[19] J.H. Su, W.Y. Chang and V.S. Tseng, Personalized Music Recommendation by Mining Social Media Tags, *Proc. of the17th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems*, 2013, pp. 291–300.

[20] A. Shepitsen, J. Gemmell, B. Mobasher and R. Burke, Personalized recommendation in social tagging systems using hierarchical clustering, *Proc. of the ACM conference on Recommender systems*, 2008, pp. 259–266.
[21] B.M. Sarwar, G. Karypis, J. Konstan and J. Riedl, Item-Based Collaborative Filtering Recommendation Algorithms, *Proc. Int. Conf. on World Wide Web*, 2001, pp. 285–295.
[22] J.H. Su, W.Y. Chang and V.S. Tseng, Personalized Music Recommendation by Mining Social Media Tags, *Proc. of the17th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES)*, Kitakyushu, Japan, 9–12 September 2013.
[23] J.H. Su, H.-H. Yeh and V.S. Tseng, A Novel Music Recommender by Discovering Preferable Perceptual-Patterns from Music Pieces, *Proc. of ACM Symposium on Applied Computing*, 2010, pp. 1924–1928.
[24] J.H. Su, H.-H. Yeh, P.S. Yu and V.S. Tseng, Music Recommendation Using Content and Context Information Mining, *IEEE Intelligent Systems* **25**(1) (2010), 16–26.
[25] K.H.L. Tso-Sutter, L.B. Marinho and L. Schmidt-Thieme, Tag-aware recommender systems by fusion of collaborative filtering algorithms, *Proc. Int. Conf. on the ACM symposium on Applied computing*, 2008, pp. 1995–1999.
[26] J. Wang, A.P. de Vries and M.J.T. Reinders, Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion, *Proc. Int. Conf. on Special Interest Group on Information Retrieval*, 2006, pp. 501–508.
[27] Yu-Xiong Wang and Yu-Jin Zhang, Nonnegative Matrix Factorization: A Comprehensive Review, *IEEE Transactions on Knowledge and Data Engineering* **25**(6) (2013), 1336–1353.
[28] J.A. Xu and K. Araki, A SVM-based Personal Recommendation System for TV programs, *Proc. Int. Conf. on Multi-Media Modeling Conference*, 2006, pp. 401–404.
[29] Z. Yu, X. Zhou, Y. Hao and J. Gu, TV Program Recommendation for Multiple Viewers Based on user Profile Merging, *User Modeling and User-Adapted Interaction* **16**(1) (2006), 63–82.
[30] Z. Zheng, H. Ma, M.R. Lyu and I. King, Collaborative Web Service QoS Prediction via Neighborhood Integrated Matrix Factorization, *IEEE Transactions on Services Computing* **6**(3) (2013), 289–299.
[31] S. Zhang, W. Wang, J. Ford and F. Makedon, Learning from Incomplete Ratings Using Non-negative Matrix Factorization, *Proc. of the Sixth SIAM International Conference on Data Mining*, 2006, pp. 549–553.