Title	An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model
Author(s)	Yoshii, K; Goto, M; Komatani, K; Ogata, T; Okuno, HG
Citation	IEEE TRANSACTIONS ON AUDIO SPEECH AND LANGUAGE PROCESSING (2008), 16(2): 435-447
Issue Date	2008-02
URL	http://hdl.handle.net/2433/50284
Right	(c)2008 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Type	Journal Article
Textversion	publisher

An Efficient Hybrid Music Recommender System Using an Incrementally Trainable Probabilistic Generative Model

Kazuyoshi Yoshii, *Student Member, IEEE*, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, *Member, IEEE*, and Hiroshi G. Okuno, *Senior Member, IEEE*

***Abstract*—This paper presents a hybrid music recommender system that ranks musical pieces while efficiently maintaining collaborative and content-based data, i.e., rating scores given by users and acoustic features of audio signals. This hybrid approach overcomes the conventional tradeoff between recommendation accuracy and variety of recommended artists. Collaborative filtering, which is used on e-commerce sites, cannot recommend nonbrated pieces and provides a narrow variety of artists. Content-based filtering does not have satisfactory accuracy because it is based on the heuristics that the user's favorite pieces will have similar musical content despite there being exceptions. To attain a higher recommendation accuracy along with a wider variety of artists, we use a probabilistic generative model that unifies the collaborative and content-based data in a principled way. This model can explain the generative mechanism of the observed data in the probability theory. The probability distribution over users, pieces, and features is decomposed into three conditionally independent ones by introducing latent variables. This decomposition enables us to efficiently and incrementally adapt the model for increasing numbers of users and rating scores. We evaluated our system by using audio signals of commercial CDs and their corresponding rating scores obtained from an e-commerce site. The results revealed that our system accurately recommended pieces including nonrated ones from a wide variety of artists and maintained a high degree of accuracy even when new users and rating scores were added.**

***Index Terms*—Aspect model, hybrid collaborative and content-based recommendation, incremental training, music recommender system, probabilistic generative model.**

I. Introduction

THE importance of music recommender systems is increasing because many online services that manage large music collections do not provide users with fully satisfactory access to their collections [1], [2]. Standard retrieval systems force users to discover their favorite musical pieces by using queries including the songs' titles or artist names. To overcome this limitation, many researchers focus on music information retrieval (MIR), which enables users to discover their favorite pieces on the basis of musical content such as genre, rhythm, and melody [3]. However, many users have difficulty in stating their musical preferences as queries. In fact, most users of music streaming services that let one freely listen to numerous songs for a flat fee, want to access their favorites one after another without querying. Recommender systems should thus be able to select musical pieces that will likely be preferred by estimating user preferences. So far, two major techniques have been proposed: collaborative and content-based filtering, and they have complementary properties.

Collaborative methods [4]–[8] recommend musical pieces to the user by considering how someone else rated them. For example, suppose that there is a target user who likes piece A. If many others like A and B, B will be recommended to the user. This technique is widely utilized in practical e-commerce services (e.g., Amazon.com and the iTunes music store) and has been demonstrated to be rather effective. However, there are two problems. The first problem is that pieces that have not been rated (e.g., newly released CDs and less well-known songs) cannot be recommended. This is known as the new-item problem or the cold-start problem. Therefore, the chances of encountering unexpected favorites are limited. The second problem is that the artists of the recommended pieces tend to be the same and are often well known because most users tend to rate highly musical pieces by the same artists. Such recommendations are unsatisfactory or meaningless.

Content-based methods [9]–[12] recommend musical pieces similar to the users' favorites in terms of musical properties. This results in a large variety of artists; i.e., various pieces are recommended even when they have not been rated. However, these methods have essential problems concerning accuracy of recommendations because similarity in content is only one of many factors characterizing user preferences. In addition, it is difficult to associate user preferences with musical content by using a real database where most users provide few rating scores. Unfortunately, reliable methods of doing this have not been established. For example, although Hoashi *et al.* [9] tried to model user preferences, their method was only verified using an artificial database where 12 subjects were asked to give rating scores. Logan [10] did not use real rating scores and instead took a set of songs in a CD album as a particular user's

Manuscript received July 4, 2007; revised September 21, 2007. This work was supported in part by the Ministry of Education, Culture, Sports, Science, and Technology (MEXT), CrestMuse Project, Japan Science and Technology Agency (JST), and a Grant-in-Aid for Scientific Research from the JSPS Research Fellowship. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Gaël Richard.

K. Yoshii, K. Komatani, T. Ogata, and H. G. Okuno are with the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan (e-mail: yoshii@kuis.kyoto-u.ac.jp; komatani@i.kyoto-u.ac.jp; ogata@i.kyoto-u.ac.jp; okuno@i.kyoto-u.ac.jp).

M. Goto is with the National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba 305-8568, Japan (e-mail: m.goto@aist.go.jp).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2007.911503

set of favorites. Celma *et al.* [11] developed a recommender system that models user preferences by collecting profiles of XML formats from the web. However, their system was not evaluated. Some studies used relevance feedback from users to improve the accuracy of recommendations [9], [12].

To solve the problems of the above techniques, we developed a hybrid recommender system that utilizes both the user's rating and musical content. Our goal was to get more accurate recommendations referring to a large variety of artists. Here, a fundamental problem is that the observed rating scores and acoustic features incompletely represent user preferences. To overcome this problem, we used a Bayesian network model called a three-way aspect model that integrates both of the observed data [13]. This model can directly represent unobservable user preferences as part of a generative mechanism for the observed data. To our knowledge, our system is the first to apply the aspect model to content-based data extracted directly from media files (audio signals) as opposite to text-based annotations of media files.

Nevertheless, a critical problem concerning the computational cost emerges in practical situations. Most academic recommender systems are based on offline methods that require costly recalculations over all the observed data when nonregistered users or new rating scores are added. To solve this problem, we developed an incremental online training method that partially updates the parameters of the three-way aspect model at low computational cost.

The rest of this paper is organized as follows. Section II specifies the requirements for the recommendations we focus on and the recommendation task. Section III reviews the conventional methods. Section IV explains our hybrid recommender system with the incremental training method. Section V reports on our experiments that used real rating scores collected from a web-shopping site, Amazon.co.jp. Section VI summarizes the key points of this study.

II. Specifications for Music Recommendation

We first describe four requirements for designing recommender systems and define the recommendation task. Note that our target users are those who want to listen to many of their favorite songs by using music streaming services that have a flat price (e.g., Last.fm and Pandora) rather than customers of web-shopping sites (e.g., Amazon.com).

A. Our Goal

We aimed at developing a recommender system that satisfies the following requirements.

1) *High recommendation accuracy*
   Given a target user, a better recommender system should select more favorite pieces and fewer disliked ones from a real database in which the number of rating scores given by users is not sufficient.
2) *Rich variety of artists*
   If the recommended pieces were performed by various artists unfamiliar to the target user, his or her chances of discovering new artists who play music that matches his or her preference would increase.
3) *Capability of recommending nonrated pieces*
   This capability enables users to find appropriate pieces that have been given a few or no rating scores. In addition, it increases the variety of artists for the recommended pieces.
4) *Prompt responses*
   If the target user adds rating scores, the recommended pieces should be reselected in real time. The recommender system should be able to deal with the increase in observed data at low computational cost.

Collaborative methods and content-based ones, which have complementary properties, cannot simultaneously satisfy the first three requirements, as discussed in Section I. We believe that advantages of these methods can be combined by using both collaborative data (rating scores) and content-based data (acoustic features). In addition, we should pay special attention to the fourth requirement.

B. Recommendation Task

An objective of music recommendation is to rank musical pieces that have not been rated by the target user. We let $U \equiv \{u_1, \ldots, u_{|U|}\}$ be users and $M \equiv \{m_1, \ldots, m_{|M|}\}$ be pieces, where $|U|$ is the number of the users and $|M|$ is that of the pieces. Here, let $u \in U$ and $m \in M$ denote variables, which are treated as probabilistic ones in the probability theory. We assumed that U and M were registered in the system in advance. Additional metadata such as titles, artist names, and genre labels are not used to make recommendations. Rating data should also be reserved in the system. In this paper, we focus on scores on a 0-to-4 scale as rating data. We let $r_{u,m}$ be a rating score given to piece m by user u, where $r_{u,m}$ is an integer between 0 and 4 (4 being the best). By collecting all the rating scores, the rating matrix, R, is obtained as

$$R = \{r_{u,m} | u \in U, m \in M\}. \tag{1}$$

When user u has not rated piece m, ϕ is substituted for $r_{u.m}$ as a symbol, representing an "empty" score for convenience. Note that most scores in R are empty because each user will have rated a few pieces in M. Collaborative methods only use R for the recommendation.

Content-based data is required to use content-based methods. We assumed that audio signals of the pieces represented by M would be available. The content of each piece is represented as a single vector of several musical elements extracted from the corresponding audio signal. Let $T \equiv \{t_1, \ldots, t_{|T|}\}$ be these elements, where $|T|$ is the number of them (dimension of content vectors). Here, let $t \in T$ also be a variable. We let $c_{m,t}$ be the value of element t in piece m. By collecting all the content vectors, the content matrix C is obtained by

$$C = \{c_{m,t} | m \in M, t \in T\}. \tag{2}$$

Given target user $u \in U$, content-based methods use C and not R but $\{r_{u,m} | m \in M\}$ for the recommendation. That is, they do not use scores given by other users in R.

III. Conventional Recommendation Methods

We describe the conventional methods that were used for the comparative experiments discussed in Section V. As mentioned in Section I, the conventional methods can be categorized

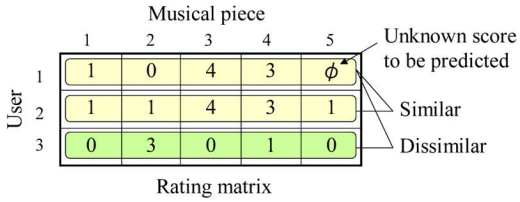

Fig. 1. Memory-based collaborative filtering method. Calculating similarities in rating-score vectors between target user and others.

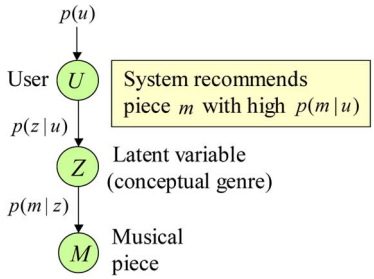

Fig. 2. Model-based collaborative filtering method. Ranking musical pieces by using an aspect model with latent variables.

into collaborative and content-based filtering. They can furthermore be categorized from the viewpoint of methodology into memory-based and model-based methods.

To make recommendations, memory-based methods operate over the entire rating matrix, R, (and content matrix C if needed). Model-based methods, in contrast, use these databases to train models that estimate user preferences, which are then used to make recommendations. In general, the latter can make prompter recommendations once the models are constructed. However, the computational cost involved in training these models tends to be high. Some studies on text-based recommendation have reported that model-based methods outperform memory-based ones in terms of recommendation accuracy [6].

A. Collaborative Filtering

We review major methods of collaborative filtering.

1) Memory-Based Methods: Typical memory-based methods try to predict the unknown rating scores of musical pieces that have not been rated by a target user, by considering someone else's scores for those pieces, as outlined in Fig. 1. That is, these methods are fundamentally based on heuristics. Given a target user u, let $\widetilde{r}_{u,m}$ be his or her predicted rating score for piece m, which is given by

$$\widetilde{r}_{u,m} = \bar{r}_u + k \sum_{\{u'|u' \neq u, u' \in U\}} w_{u,u'}(r_{u',m} - \bar{r}_{u'}) \qquad (3)$$

where $\bar{r}_u$ is the average rating score of user u and $\bar{r}_{u'}$ is that of user u'. $w_{u,u'}$ is a weight that reflects the preference similarity between users u and u', and k is a normalizing factor so that the absolute values of the weights add up to unity. That is, $\sum_{u'} |w_{u,u'}| = 1$. After the score is predicted, pieces are ranked according to $\widetilde{r}_{u,m}$.

Several measures are used to calculate similarity. The most popular is the Pearson correlation coefficient [5] with which similarity is defined as

$$w_{u,u'} = \frac{\sum_m (r_{u,m} - \bar{r}_u) \sum_m (r_{u',m} - \bar{r}_{u'})}{\sqrt{\sum_m (r_{u,m} - \bar{r}_u)^2 \sum_m (r_{u',m} - \bar{r}_{u'})^2}} \qquad (4)$$

where summations over m are for pieces rated by both u and u'. However, there are usually very few of those pieces when the rating matrix R is sparse. Therefore, this basic similarity calculation often fails.

To solve this problem, empty scores in R are replaced with a default score r_D. We empirically set the value for r_D to 2.5, which is biased (cf., a neutral score is 2 on the 0-to-4 scale), because most users tend to give high scores (3 and 4) more often than low ones (0 and 1).

2) Model-Based Methods: From the viewpoint of the probability theory, the unknown rating scores of a target user can be predicted by calculating their expected values

$$\widetilde{r}_{u,m} = E(r|u,m) = \sum_r r \times p(r|u,m) \qquad (5)$$

where $r \in \{0, \ldots, 4\}$ is a probabilistic variable, and $p(r|u,m)$ is the probability of score r when user u and piece m are observed. This probability is estimated by using actual rating scores in R by assuming probabilistic models [6]. Note that these methods are based not on some ad hoc heuristic rules but on a statistical model learned from the underlying data using machine learning techniques.

An alternative approach is to rank musical pieces for given user u according to $p(m|u)$, which is the conditional probability of piece m when user u is observed. Here, we assume that the co-occurrence of piece m and user u (e.g., events that user u listens to piece m) is likely to be observed if user u prefers piece m, i.e., the joint probability, $p(u,m)$, increases as score $r_{u,m}$ or $\widetilde{r}_{u,m}$ becomes large. Therefore, the probability $p(m|u) \propto p(u,m)$ represents how likely user u is to prefer piece m. To estimate $p(m|u)$, Hofmann and Puzicha [14] used a probabilistic generative model called an aspect model. A data mining technique based on the aspect model is known as the probabilistic latent semantic analysis (pLSA) [15].

The aspect model introduces multiple latent variables, which represent conceptual genres, as outlined in Fig. 2. Let $Z \equiv \{z_1, \ldots, z_{|Z|}\}$ be these variables, where $|Z|$ is the number of them. An interpretation of this model is that user u stochastically selects conceptual genre z according to his or her preference $p(z|u)$, and then z stochastically generates piece m according to the probability $p(m|z)$. The conceptual genres, which do not correspond to "genres" in the general sense, are not given in advance. They are automatically determined so that the model provides the best explanation of the generative mechanism for the observed rating scores. To put this more concretely, $p(z|u)$ and $(m|z)$ are statistically estimated by using the EM algorithm [16] (the method of estimating parameters is a simplified version of ours described in Section IV). After the parameters are estimated, musical pieces are ranked for each user u according to $p(m|u) \propto \sum_z p(m|z)p(z|u)$.

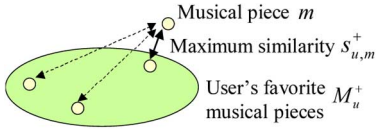

Fig. 3. Memory-based content-based filtering method. Calculating similarities in musical content between the user's favorites and other pieces.

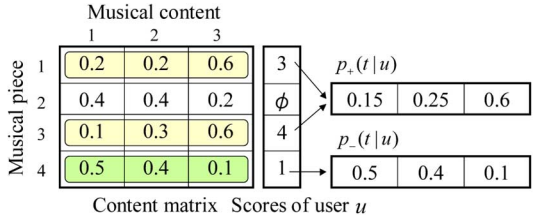

Fig. 4. Model-based content-based filtering method. Estimating the preferences of the target user by using his or her rating scores.

B. Content-Based Filtering

We review major methods of content-based filtering.

1) Memory-Based Methods: Typical memory-based methods try to rank musical pieces on the basis of the similarity of musical content by representing user preferences in a musical-content space. Let $\boldsymbol{c}_m \equiv (c_{m,1}, \ldots, c_{m,|T|})$ denote a content vector of piece m. Let $M_u^+ \equiv \{m | r_{u,m} = 3, 4\}$ be a set of pieces that were given positive scores (3 or 4) by user u. In the same way, let $M_u^- \equiv \{m | r_{u,m} = 0, 1\}$ be a set of pieces that were given negative scores (0 or 1). Given target user u, the algorithm is

1) If M_u^+ is not empty, the set of content vectors $C_u^+ : \{\boldsymbol{c}_m | \, m \in M_u^+\}$ represents the musical taste of u. If M_u^- is not empty, the set of vectors, $C_u^- : \{\boldsymbol{c}_m | m \in M_u^-\}$ represents the musical dislikes of u.
2) The similarities between content vectors in C_u^+ and the content vector $\boldsymbol{c}_m$ $(r_{u,m} = \phi)$ can be calculated with a similarity measure, as outlined in Fig. 3. Let $s^+_{u,m}$ be the maximum similarity, which indicates how likely u will prefer m. Then, $s^+_{u,m}$ is calculated for each piece m. If M_u^+ is empty, $s^+_{u,m}$ is set to zero.
3) Let $s^-_{u,m}$ be the maximum similarity between content vectors in C_u^- and content vector $\boldsymbol{c}_m$ $(r_{u,m} = \phi)$, which is calculated in the same way in the previous step. If M_u^- is empty, $s^-_{u,m}$ is set to zero.
4) The musical pieces $\{m | r_{u,m} = \phi\}$ that have not been rated by u are ranked according to the total value $s_{u,m}$, which is

$$s_{u,m} = s^+_{u,m} - s^-_{u,m}. \tag{6}$$

The cosine measure is often used to calculate the similarities between two vectors [9]. Note that if u only provides neutral scores (2), random pieces are recommended.

2) Model-Based Methods: Recommending musical pieces can be viewed as categorizing them into two classes: favorites and disliked. This is a standard machine-learning problem. Naïve Bayes models have often been used to categorize text documents (e.g., spam-mail filtering) [17]. However, one problem is that we should define and calculate musical elements which correspond to words in text documents. We propose a novel method of solving this problem in Section IV-B. Once the content matrix C is obtained with our method, the binary categorization models are given by

$$p_+(m|u) = \prod_t p_+(t|u)^{c_{m,t}} \tag{7}$$

$$p_-(m|u) = \prod_t p_-(t|u)^{c_{m,t}} \tag{8}$$

where $p_+(m|u)$ $(p_-(m|u))$ indicates how likely user u is to like (dislike) piece m. $p_+(t|u)$ $(p_-(t|u))$ indicates how likely u is to like (dislike) element t, which is trained as outlined in Fig. 4

$$p_+(t|u) = \sum_{m \in M_u^+} c_{m,t}, \quad p_-(t|u) = \sum_{m \in M_u^-} c_{m,t}. \tag{9}$$

Finally, musical pieces are ranked for each u according to the total value $p_{u,m}$ which is obtained by

$$p_{u,m} = \frac{p_+(m|u)}{p_-(m|u)}. \tag{10}$$

If M_u^+ (M_u^-) is empty, $p_+(m|u)$ $(p_-(m|u))$ is set to one for convenience.

IV. Hybrid Recommender System

We propose a hybrid music recommender system that theoretically integrates collaborative data (rating scores of users) and content-based data (acoustic features of audio signals) to meet the four requirements described in Section II-A. First, we will discuss the problems with integrating both data. Next, we will explain our elegant approach based on a probabilistic generative model that can be incrementally trained.

A. Problems

We need to solve two problems to design a hybrid recommender system. Each problem is discussed below.

1) Reliable Integration: The first problem is to reflect the collaborative and content-based data when making recommendations. An easy solution is to use collaborative and content-based methods in parallel [18]–[20] or in cascade [21]–[25]. However, such an approach has drawbacks. Although meta recommender systems have been proposed to select a recommender system among conventional ones on the basis of certain quality measures [18], [19], the disadvantages of the selected system are inherited. Moreover, the heuristics-based integration dealt with in other studies lacks a principled justification. For example, Claypool *et al.* [20] proposed a linear combination of rating scores predicted by several conventional systems. Typical cascade systems [21]–[23] first represent user preferences by using content-based data and then make recommendations in a collaborative way that calculates the similarities of the content-based user preferences. Melville *et al.* [24] used a collaborative method after predicting unknown rating scores by using content-

based data. A similar content-boosted approach was proposed by Hayes [25]. A principled way of integration is to take a model-based approach, which statistically estimates user preferences on the basis of a unified model. Here, note that the observed data (rating scores and acoustic features) are incomplete; i.e., these data partially represent the latent user preferences.

2) Efficient Calculation: The second problem, which has been scarcely dealt with, is to efficiently adapt a recommender system according to the increase in rating scores and users. An easy solution is to take a memory-based approach, which is originally free from this problem because the whole data is always used to make recommendations. However, this results in the late responses. Yu *et al.* [26] tried to overcome this disadvantage by using a probabilistic method in a pure collaborative filtering context. On the other hand, Zhang *et al.* [27] proposed an efficient method that incrementally trains an aspect model used for model-based collaborative filtering. To our knowledge, there are no studies on incremental adaptation of hybrid recommender systems. We need to carefully design a hybrid architecture while considering whether the previous prominent methods can be applied or not.

B. Our Approach

To solve the problems, we take the following strategies:

1) Model-Based Integration: We use a probabilistic generative model, called a three-way aspect model, proposed by Popescul *et al.* [13]. This model is an extended version of the aspect model (pLSA) proposed by Hofmann [15] (see Section III-AII), and it explains the probabilistic generative mechanism for the observed data (rating scores and acoustic features) by introducing a set of latent variables. As part of the generative mechanism, the model directly represents user preferences (latent favorite genres), which are statistically estimated with a theoretical proof. This estimation makes the recommendations more reliable.

2) Incremental Training: We propose a method that incrementally adapts the three-way aspect model on the basis of an extended version of Zhang's method [27].

C. Model-Based Integration Method

We will discuss how the three-way aspect model is applied to music recommendation. After that, we explain its implementation.

1) Analogy to Document Recommendation: Popescul's hybrid model cannot be directly applied to our system because it was designed for recommending text documents. The document content is represented on the basis of the "bag-of-words" model originally proposed in the field of language processing, i.e., the content of a document is represented as a set of frequencies of informative words.

To apply the three-way aspect model to music recommendation, the content of each piece should be represented as a single vector in which all dimensions are semantically equivalent. For example, each dimension always represents a word frequency with Popescul's method. In addition, all dimensions of each vector should add up to unity.

2) Application to Music Recommendation: We propose a "bag-of-timbres" model in analogy with the bag-of-words model to meet the above-mentioned conditions. The content of each piece is represented as a bag of multiple timbres, i.e., a set of weights of *polyphonic timbres*. Aucouturier *et al.* [28] proposed the original concept of polyphonic timbres, which do not represent the perceptual "sounds" of individual instruments but of their combinations (mixed sounds). These features are important factors that characterize the textures of musical pieces. In addition, polyphonic timbres can easily be extracted from various audio signals because the instrument parts do not need to be separated (this separation is quite difficult). However, Aucouturier *et al.* pointed out that this approach has a performance limitation on timbral-similarity-based audio clustering. In contrast, we expect that this approach works well by integrating it with the "wisdom-of-crowds" approach based on collaborative data.

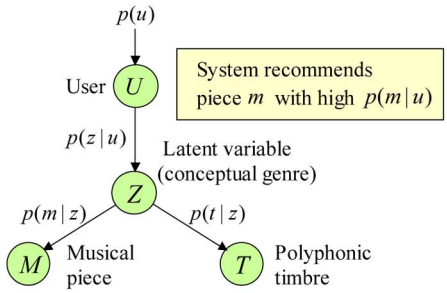

Fig. 5. Asymmetric representation of our aspect model using polyphonic timbre weights as music content.

3) Three-Way Aspect Model Based on Bags of Timbres: The observed data in the three-way aspect model are associated with latent variables $Z \equiv \{z_1, \ldots, z_{|Z|}\}$, where $|Z|$ is the total number, as outlined in Fig. 5. Each latent variable corresponds to a conceptual genre. Given a target user u, a set of conditional probabilities $\{p(z|u)|z \in Z\}$ reflects the musical taste of u. One possible interpretation is that user u stochastically selects conceptual genre z according to his or her preference, $p(z|u)$, and then z stochastically generates piece m and polyphonic timbre t according to their probabilities $p(m|z)$ and $p(t|z)$. We assumed the conditional independence of users, pieces, and timbres through the latent genres. In this model, all users and all musical pieces can be observed for selection of any genre, whereas most clustering methods assign each user and each piece to a single genre class. To deal with the increase in rating scores and users, we only need to update user preferences $\{p(z|u)|z \in Z\}$.

4) Calculation of Bags of Timbres: To calculate bags of timbres from audio signals, we used Mel-frequency cepstral coefficients (MFCCs), which have been used in many studies on genre classification [29]. Their method was used to apply a Gaussian mixture model (GMM) to the MFCCs extracted from each musical piece. The similarity between two pieces was measured as the reciprocal of the distance between corresponding GMMs that was obtained by sampling.

We also built a GMM for MFCCs extracted from each piece. That is, we obtained $|M|$ GMMs in total. We assumed that each Gaussian in a GMM would represent the MFCC distribution of a particular polyphonic timbre, i.e., the mixture weights of

Gaussians would correspond to the weights of timbres. Unfortunately, we could not use Aucouturier's method because the Gaussians in one GMM were different from those in the other GMMs. Therefore, each GMM represented a different combination of polyphonic timbres.

Our unique idea to solve this problem was to let the bags of timbres of all the pieces share the same combination of Gaussians. The means and covariances of the Gaussians were estimated by using numerous MFCCs extracted *not individually from each piece but from all the pieces*, and the mixture weights of the Gaussians were discarded in this estimation. The weights of the polyphonic timbres in each piece were obtained as the mixture weights of the fixed Gaussians in that piece; only the mixture weights were reestimated by using the MFCCs of the single piece.

First, 13-dimensional MFCCs were extracted from audio signals sampled at 16.0 kHz by applying short-time Fourier transformation (STFT) with a Hanning window of 200 ms. The shifting interval was 100 ms. Then, 28-dimensional feature vectors were obtained (MFCCs, energy, and their delta components). We let $\{\boldsymbol{f}_{m,i}|1 \leq i \leq I_m\}$ be feature vectors extracted from piece m, where I_m is the total number. Next, the parameters of the Gaussians were estimated for all the pieces by using the expectation-maximization (EM) algorithm [16], where the number of mixtures was set to 64. That is, $|T| = 64$.

The number of mixtures $|T|$, which corresponds to the GMM complexity, influenced the recommendation accuracy defined in Section V-B. We empirically found that $|T| = 64$ was sufficient because the recommendation accuracy did not almost improve much if we took $|T|$ to be any higher.

Here, let N_t be the tth Gaussian, which is given by

$$N_t(\boldsymbol{x}) = \frac{1}{(2\pi)^{(28/2)}|\Sigma_t|^{(1/2)}} \exp\left(-\frac{1}{2}D^2(\boldsymbol{x}, \boldsymbol{\mu}_t)\right) \quad (11)$$

where $\boldsymbol{\mu}_t$ is the mean vector and Σ_t is the covariance matrix. D^2 is the squared Mahalanobis distance given by

$$D^2 = (\boldsymbol{x} - \boldsymbol{\mu}_t)^T \Sigma_t^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_t). \quad (12)$$

The content value, $c_{m,t}$, which is the weight of timbre t in piece m, is obtained by

$$c_{m,t} = \frac{1}{I_m} \sum_{i=1}^{I_m} l_{m,i} N_t(\boldsymbol{f}_{m,i}) \quad (13)$$

where $l_{m,i}$ is a normalizing factor so that $\sum_t N_t(\boldsymbol{f}_{m,i}) = 1$.

5) Formulation of Three-Way Aspect Model: We explain the mathematical formulation for the three-way aspect model. Let a tuple, (u, m, t), be the co-occurrence of three probabilistic variables $u \in U$, $m \in M$, and $t \in T$. Each tuple, (u, m, t), corresponds to observations where user u listens to timbre t in piece m. Let $n(u, m, t)$ be the number of these observations. In this paper, we assume that $n(u, m, t)$ is proportional to the product of $r_{u,m}$ and $c_{m,t}$

$$n(u, m, t) \propto r_{u,m} \times c_{m,t}. \quad (14)$$

Recall that in the definitions of $r_{u,m}$ and $c_{m,t}$

- $r_{u,m}$ is the rating score of user u for piece m. A default rating score (2.5) was substituted for the empty scores in our method, as described in Section III-A.
- $c_{m,t}$ is the weight of polyphonic timbre t in piece m.

This assumption is based on the general fact that (u, m, t) co-occurs more frequently if user u prefers piece m more or the weight of timbre t in piece m is higher.

In the same way, a tuple (u, m, t, z) is defined as the co-occurrence of four variables $u \in U$, $m \in M$, $t \in T$, and $z \in Z$. Each tuple, (u, m, t, z), corresponds to *unobservable events*, where user u selects genre z, and then genre z simultaneously generates timbre t and piece m. Let $p(u, m, t, z)$ be the probability of the co-occurrence (u, m, t, z). The assumed conditional independence over U, M, and T through Z leads to a symmetric form of $p(u, m, t, z)$ as follows:

$$p(u, m, t, z) = p(u)p(z|u)p(m|z)p(t|z) \quad (15)$$

$$= p(z)p(u|z)p(m|z)p(t|z) \quad (16)$$

where $p(u)$ is the prior probability of user u. Marginalizing z, we obtain the joint probability distribution $p(u, m, t)$ over U, M, and T

$$p(u, m, t) = \sum_z p(z)p(u|z)p(m|z)p(t|z) \quad (17)$$

where $p(z)$ is the prior probability for genre z, and $p(u|z)$ is the probability that genre z will generate user u.

The unknown model parameters are $\{p(z)|z \in Z\}$, $\{p(u|z)|u \in U, z \in Z\}$, $\{p(m|z)|m \in M, z \in Z\}$, and $\{p(t|z)|t \in T, z \in Z\}$, which should be estimated by using the rating matrix R and content matrix C. After they are estimated, the musical pieces are ranked for a given user u according to $p(m|u) \propto \sum_t p(u, m, t)$.

6) Estimation of Model Parameters: We explain the method of estimating the model parameters using the EM algorithm [16]. Here, we assume that each event occurs independently. The likelihood of the parameters for the observed data is given by

$$L' = \prod_{u,m,t} p(u, m, t)^{n(u,m,t)} \quad (18)$$

$$= \prod_{u,m,t} \left(\sum_z p(z)p(u|z)p(m|z)p(t|z)\right)^{n(u,m,t)}. \quad (19)$$

Given the observed data (rating matrix R and content matrix C), the log-likelihood, L, is obtained as

$$L = \sum_{u,m,t} n(u, m, t) \log p(u, m, t). \quad (20)$$

We use the EM algorithm [16] to estimate the parameters so that log-likelihood L reaches a local maximum as follows:

E step

$$p(z|u, m, t) = \frac{(p(z)p(u|z)p(m|z)p(t|z))^{\beta}}{\sum_{z'} (p(z')p(u|z')p(m|z')p(t|z'))^{\beta}} \quad (21)$$

M step

$$p(u|z) \propto \sum_{m,t} n(u,m,t)p(z|u,m,t) \tag{22}$$

$$p(m|z) \propto \sum_{u,t} n(u,m,t)p(z|u,m,t) \tag{23}$$

$$p(t|z) \propto \sum_{u,m} n(u,m,t)p(z|u,m,t) \tag{24}$$

$$p(z) \propto \sum_{u,m,t} n(u,m,t)p(z|u,m,t) \tag{25}$$

where we have introduced a parameter β that is set to one in the basic EM algorithm. These steps are iterated alternately until L converges to the local maximum. Note that the local maximum problem cannot be ignored in practice because the matrices R and C are sparse.

To solve this sparseness problem, we use the deterministic annealing EM algorithm (DAEM) [30], which is a variant of the EM algorithm. The DAEM algorithm is equivalent to the tempered EM algorithm proposed by Hofmann [15] and was used for training Popescul's three-way aspect model [13]. β works as an inverse computational temperature, which is gradually increased. The DAEM algorithm starts with $\beta = \beta_{\min}$ and increases β with ratio η, using $\beta \rightarrow \eta \times \beta$ when log-likelihood L converges. Finally, β is set to one. It is sufficient to set $\beta_{\min}$ to 0.1 and η to 1.2 in practice.

The computational complexity of training the model with this method is $O(|U||M||T||Z|) \approx O(|U||M|)$, considering that $|T| = 64$ and $|Z| = 10$ remain constant and the number of users and number of musical pieces are not fixed. This means that both computational time and memory use increase rapidly according to $O(|U||M|)$. One possible solution is to categorize users and musical pieces into fewer groups. This will be done in a future work.

D. Incremental Training Method

To achieve prompt responses, we propose an efficient incremental training method that partially updates the parameters of the three-way aspect model. Ours is an extended version of Zhang's method [27], which was designed for incrementally training a topologically different aspect model. After this, we will call the three-way aspect model that was initially obtained using EM-based training a *base model*. We will call a model that was obtained by incrementally training the base model an *updated model*.

Our method individually addresses the following two cases to obtain the updated model.

1) Recommendation given to a registered user ($\in U$) who provides new rating scores.
2) Recommendation given to a nonregistered user ($\notin U$) who provides some rating scores.

While the size of the model (the number of parameters) remains unchanged in the first case, it increases in the second because nonregistered users are added. Our method differs from typical incremental training methods based on a fixed model size in this regard. Next, we will explain the incremental training algorithms for both bases.

1) Updating Profiles of Registered Users: Given a specific user u, the conditional probability distribution $\{p(z|u)|z \in Z\}$, which is called a *user profile*, captures his or her musical preferences. Recall that $p(z|u)$ represents how likely user u is to select conceptual genre z according to his or her musical preferences. The three-way aspect model assumes that the profiles of all users are independent, as outlined in Fig. 5. Therefore, when a user gives new rating scores, we only need to update his or her profile without affecting the profiles of others to keep the log-likelihood maximized. This results in lower computational cost.

We aim at updating a profile of user u': $\{p(z|u')|z \in Z\}$, where $u' \in U$ is a registered user who gives new rating scores. We assume that model parameters other than the profile of user u' are constant. Therefore, maximizing the log-likelihood L is equivalent to maximizing the sum of terms including user u' in L. We let $L_{u'}$ be the log-likelihood for the observed data concerning user u', as follows:

$$L_{u'} = \sum_{m,t} n(u',m,t) \log p(m,t|u') \tag{26}$$

$$= \sum_{\langle m,t|u' \rangle} \log p(m,t|u') \tag{27}$$

where we have introduced a new operator $\sum_{\langle m,t|u' \rangle}$ for X (X is an arbitrary value), which represents $\sum_{m,t} n(u',m,t)X$. Using Jensen's inequality, we can rewrite (27) as

$$L_{u'} = \sum_{\langle m,t|u' \rangle} \log \sum_z p(m|z)p(t|z)p(z|u') \tag{28}$$

$$= \sum_{\langle m,t|u' \rangle} \log \sum_z \frac{p(m|z)p(t|z)}{\delta_{m,t}} p(z|u')\delta_{m,t} \tag{29}$$

$$\geq \sum_{\langle m,t|u' \rangle} \sum_z \frac{p(m|z)p(t|z)}{\delta_{m,t}} \log p(z|u') + \sum_{\langle m,t|u' \rangle} \log \delta_{m,t} \tag{30}$$

where $\delta_{m,t}$ is a supplementary function given by

$$\delta_{m,t} = \sum_z p(m|z)p(t|z). \tag{31}$$

Because $p(m|z)$ and $p(t|z)$ are almost constant, maximizing $L_{u'}$ is approximately equivalent to maximizing the first term of (30). Therefore, we can obtain the following maximization problem:

$$\text{maximize } \tilde{L_{u'}} = \sum_{\langle m,t|u' \rangle} \sum_z \frac{p(m|z)p(t|z)}{\delta_{m,t}} \log p(z|u') \tag{32}$$

$$\text{s.t.} \sum_z p(z|u') = 1 \tag{33}$$

where $\tilde{L_{u'}}$ is a constrained objective function and (33) is a constraint function. Here, we use the Lagrange multiplier method [31] to solve this problem. Introducing an unknown multiplier λ, we define $L^*_{u'}$ as

$$L^*_{u'} = \tilde{L_{u'}} - \lambda \left(\sum_z p(z|u') - 1 \right). \tag{34}$$

The partial derivative of $L^*_{u'}$ with respect to $p(z|u')$ is given by

$$\frac{\partial L^*_{u'}}{\partial p(z|u')} = \frac{\sum_{\langle m,t|u'\rangle} \frac{p(m|z)p(t|z)}{\delta_{m,t}}}{p(z|u')} - \lambda. \tag{35}$$

To maximize $\tilde{L_{u'}}$, (35) should be zero. Therefore, we obtain

$$\lambda p(z|u') = \sum_{\langle m,t|u'\rangle} \frac{p(m|z)p(t|z)}{\delta_{m,t}}. \tag{36}$$

Substituting (36) into (33) gives

$$\sum_z p(z|u') = 1 \Leftrightarrow \lambda = \sum_z \sum_{\langle m,t|u'\rangle} \frac{p(m|z)p(t|z)}{\delta_{m,t}} \tag{37}$$

$$\Leftrightarrow \lambda = \sum_{\langle m,t|u'\rangle} \frac{\sum_z p(m|z)p(t|z)}{\delta_{m,t}} \tag{38}$$

$$\Leftrightarrow \lambda = \sum_{m,t} n(u',m,t). \tag{39}$$

Substituting (39) into (36), we finally obtain the updating formula

$$p(z|u') = \frac{\sum_{m,t} n(u',m,t) \frac{p(m|z)p(t|z)}{\sum_{z'} p(m|z')p(t|z')}}{\sum_{m,t} n(u',m,t)}. \tag{40}$$

The computational complexity of updating the profile of user u' is $O(\Delta|M|)$, where $\Delta|M|$ is the number of pieces that were newly rated by user u'. To update the profile, we only need to recalculate the $\Delta|M|$ terms concerning these pieces in each summation of the updating formula (40).

2) Creating Profiles of Nonregistered Users: We aim at creating a profile of user u': $\{p(z|u')|z \in Z\}$, where $u' \notin U$ is a nonregistered user who has some rating scores $\{r_{u',m}|m \in M\}$. Note that these scores were not used for training the base model. We can use the updating formula (40) in this case to create the profile by using $p(m|z)$ and $p(t|z)$, which were estimated for the rating scores of registered users U.

V. Evaluation

We will report on several experiments that were conducted to determine whether our hybrid recommender system satisfies the four requirements described in Section II-A. First, we compared our method based on the three-way aspect model with the four conventional methods described in Section III in terms of recommendation accuracy. Next, we evaluated our method in terms of variety of recommended artists and capability of recommending nonrated pieces. Finally, we evaluated our incremental training method in terms of recommendation accuracy.

A. Experimental Conditions

It is ideal to use large-scale rating data in which the number of rating scores given by users is sufficient to conduct reliable comparative experiments, However, collecting rating scores based on questionnaires is extremely time consuming. In addition, the ratio of negative scores tends to be much higher in artificial rating data (e.g., the rating data in the previous study [9]) than in real rating data. Users tend to voluntarily give much more positive scores than negative scores.

TABLE I
Composition of Rating Scores

Score	4	3	2	1	0	Total
# (scores)	1433	473	212	120	236	2474
Ratio	57.9%	19.1%	8.57%	4.85%	9.54%	100%

To deal with this problem, we collected real rating scores from web sites [32]. Amazon.co.jp provides application programming interfaces (APIs) that allow us to download almost all information on a web site.[1]

The musical pieces we used were Japanese songs on single CDs that were ranked in weekly top-20 sales rankings from April 2000 to December 2005. The corresponding scores with user IDs were collected from Amazon.co.jp. If a user has rated multiple pieces, we can identify the scores given by the same user. However, there were many unreliable users and pieces that had few/no scores. To deal with this problem, we selected users and pieces so that the number of scores given by a user and the number of scores given to a piece were always more than 4. As a result, $|U|$ was 316 and $|M|$ was 358. Table I lists the composition of actual scores in the rating matrix R. The density of R was 2.19%, which was almost equal to the density in the previous study [6]. This means R contains practical data.

By using the prepared rating data, we compared our hybrid method based on the three-way aspect model with the four conventional methods described in Section III:

1) Memory-based collaborative method using the Pearson correlation coefficient (called *memory-CF*).
2) Model-based collaborative method using Hofmann's aspect model (called *model-CF*).
3) Memory-based content-based method using the cosine distance measure (called *memory-CB*).
4) Model-based content-based method using the naïve Bayes models (called *model-CB*).

Recall that $|T|$ (the number of polyphonic timbres) was 64. $|Z|$ (the number of latent variables) was set to 10.

B. Evaluation Measure

The experiments were conducted with tenfold cross validation; i.e., a training matrix R_t and an evaluation matrix R_e were created from the rating matrix R by randomly masking 10% of the actual scores in R, as outlined in Fig. 6. The five methods, including ours, were used to rank musical pieces for each user by using R_t and the content matrix C, if needed.

We devised an evaluation measure to calculate the accuracy of recommendation that focuses on the ratio of favorite pieces to recommended pieces whose scores are masked. We examined the entire top-x rankings of all users ($x = 1, 3, 10$). Fig. 7 shows an example for the case of $x = 3$. Note that we could not evaluate all the recommended pieces (the total number was $x|U|$) because most of them had not actually been rated by users (the corresponding scores were ϕ in R_e). Here, we let N_r be the *total number of recommended pieces* whose scores were masked but

[1][Online]. Available: http://www.amazon.com/gp/aws/landing.html.

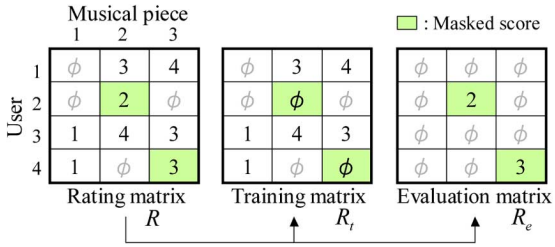

Fig. 6. Preparation of tenfold cross validation. Creating training matrix R_t and evaluation matrix R_e by randomly masking 10% of the actual scores in rating matrix R.

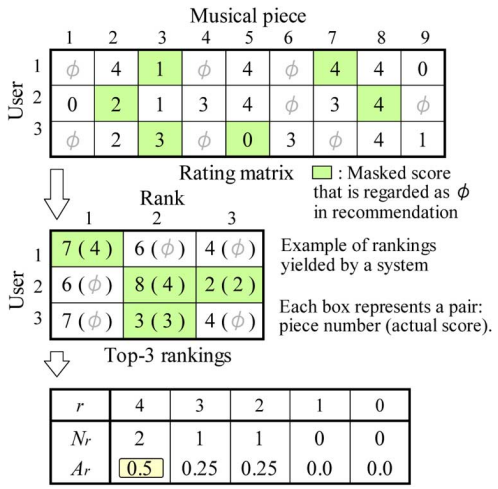

Fig. 7. Evaluation of recommendation accuracy. Calculating ratio of recommended pieces that were given highest actual scores.

were actually r $(0 \leq r \leq 4)$ and let N be $N = \sum_r N_r$. Obviously, N was much less than $x|U|$. We let A_r be the ratio of N_r to N, i.e.,

$$A_r = \frac{N_r}{N}. \tag{41}$$

A higher value for A_4 and a lower value for A_0 indicate better performance. Note that A_r is not equal to 20% even when recommending random pieces. In this case, $A_4, \ldots, A_0$ are 57.9%, 19.1%, 8.57%, 4.85%, and 9.54%, respectively. They are equal to the ratios listed in Table I.

The precision and recall rates, which are commonly used to evaluate document recommender systems, were not suitable for our task. Instead, we let N'_r be the *total number of pieces* whose scores were masked but were actually r. In the case of Fig. 7, $N'_4, \ldots, N'_0$ are 2, 1, 1, 1, and 1, respectively. The precision and recall rates, P_r and R_r, are given by

$$P_r = \frac{N_r}{x|U|}, \quad R_r = \frac{N_r}{N'_r}. \tag{42}$$

If we used these measures, higher values for P_4 and R_4 would indicate better performance. Obviously, P_4 and R_4 are proportional to N_4 because the denominators are constant. Therefore, a way of improving performance is to increase N_4, i.e., to recommend more pieces whose scores were masked but were actually rated. However, such recommendations are meaningless. In addition, N_1 and N_0 are not related to P_4 and R_4. This means the disliked pieces in the recommended ones are not taken into account.n

TABLE II
RECOMMENDATION ACCURACY: A_4

Rankings x	Our method	Memory CF	Model CF	Memory CB	Model CB
1	93.5%	77.0%	95.2%	78.4%	80.0%
3	86.4%	75.9%	77.5%	79.5%	66.7%
10	80.7%	71.1%	74.2%	71.5%	71.9%

TABLE III
RECOMMENDATION INACCURACY: A_0

Rankings x	Our method	Memory CF	Model CF	Memory CB	Model CB
1	0.00%	3.54%	0.00%	0.00%	0.00%
3	0.00%	3.38%	5.00%	0.86%	8.33%
10	1.24%	3.47%	3.33%	4.30%	7.02%

C. Evaluation of Recommendation Accuracy

The results of the comparative experiments revealed that our hybrid method outperformed the four conventional methods in terms of recommendation accuracy A_4 (Table II). For $x = 3, 10$, the accuracies obtained with our method were much higher than those with the other methods. For $x = 1$, although the accuracy of our method (93.5%) was slightly less than the best accuracy obtained by model-CF (95.2%), the difference was small. Note that our experiments with rating scores collected from Amazon.co.jp were advantageous to collaborative methods. Users tend to give positive scores (3 and 4) to most musical pieces performed by their favorite artists. It is more effective to focus on positive scores than on acoustic features to deal with this tendency. When we increased x, the accuracy of our method gradually deteriorated $(93.5\% \rightarrow 86.4\% \rightarrow 80.7\%)$ while the accuracy of model-CF more rapidly deteriorated $(95.2\% \rightarrow 77.5\% \rightarrow 74.2\%)$. This indicates that acoustic features are important factors that characterize user preferences, although artist names are the most dominant factors at Amazon.co.jp. In the future, we plan to check whether music streaming services that have flat pricing have this feature.

Our method yielded the lowest recommendation inaccuracy A_0 (Table III). Our method recommended the fewest pieces that were actually hated by users. This is an important aspect for ensuring stress-free environment.

We confirmed that our bag-of-timbres model worked well for representing musical content derived from audio signals. The accuracies achieved by memory-CB were slightly better than those by memory-CF. These results also proved the importance of acoustic features in modeling user preferences. However, as we pointed out in Sections I and IV-C2, only focusing on the content-based aspect of user preferences limits the recommendation accuracy improvement.

D. Evaluation of Artist Variety

We propose two measures to calculate the variety of artists. Artists who perform musical pieces correspond to the authors of documents and directors of movies in other recommendation tasks. However, variety has not been investigated in the field of text-based recommendation. Our study is the first to examine the variety of artists. Given a target user u, we let v_A be the number of artists in x pieces that were recommended to user u $(v_A \leq x)$. We then let v_M be the number of recommended pieces by new

TABLE IV
RATIO OF NUMBER OF ARTISTS: V_A/x

Rankings x	Our method	Memory CF	Model CF	Memory CB	Model CB
1	1.00	1.00	1.00	1.00	1.00
3	0.930	0.833	0.843	0.950	0.807
10	0.854	0.760	0.903	0.842	0.565

TABLE V
PERCENTAGE OF PIECES BY NEW ARTISTS: V_M/x

Rankings x	Our method	Memory CF	Model CF	Memory CB	Model CB
1	90.8%	62.9%	93.6%	87.1%	97.1%
3	91.3%	69.3%	94.7%	85.3%	97.3%
10	92.4%	80.0%	95.1%	87.8%	95.7%

artists whose pieces had not been rated by user u ($v_M \leq x$). We let V_A be the average for v_A and V_M be that for v_M over all users. Obviously, V_A and V_M were less than x. Higher values of V_A and V_M indicate a larger variety of artists.

The results showed that our method recommended a sufficiently diverse variety of artists (Tables IV and V). Note that the values listed in the tables are normalized by the number of recommended pieces x in order to compare all results directly. The values of V_A gotten with our method were the second best for $x = 3, 10$. The values of V_M were the third best. However, we think that the differences in measures between ours and the best methods were small. More important is the simultaneous achievement of high recommendation accuracy and rich variety.

We proved that memory-CF, which is used in many e-commerce services, provides only a limited variety of artists. V_A and V_M for memory-CF were the lowest for $x = 1, 3, 10$. V_M was especially small (0.629) in the case of $x = 1$. This means the probability that a user will discover an unknown artist with a top-1 ranked piece is at most 62.9%, which is much lower than the probabilities obtained with the other methods (about 90%). On the other hand, V_A and V_M were generally good for the content-based methods. However, many disliked pieces performed by various artists contributed to enriching variety. Indeed, these methods were inferior to the others in terms of recommendation accuracy, as listed in Tables II and III.

E. Evaluation of Capability of Recommending Nonrated Musical Pieces

To evaluate the recommendation accuracy for nonrated musical pieces, we did another tenfold cross validation by masking actual scores of 10% of M; i.e., the training matrix R_t included $(1/10)|M|$ nonrated pieces and the evaluation matrix R_e included the actual scores for these pieces. The results revealed that our method could make reasonable recommendations for nonrated pieces (Table VI). It competed with the content-based methods in terms of accuracy of recommending favorite pieces (the values for A_4+A_3), although A_4 was not high. Its probabilities of recommending disliked pieces were similar to those of the conventional methods. Note that the recommended nonrated pieces amounted to less than 5% of that of all recommended pieces.

TABLE VI
CAPABILITY OF RECOMMENDING NONRATED PIECES

Rankings x	favorites A_4	favorites A_3	dislikes A_1	dislikes A_0
1	65.8%	23.0%	1.02%	4.60%
3	65.0%	27.5%	0.00%	2.50%
10	67.9%	21.2%	1.09%	4.89%

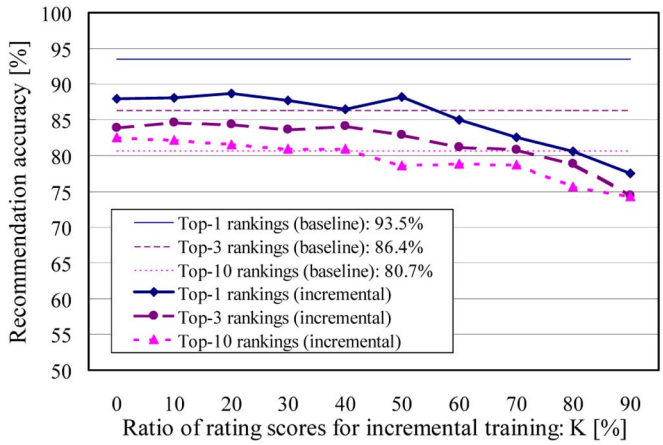

Fig. 8. Recommendation accuracy A_4 versus the rating scores for incremental training.

F. Evaluation of Incremental Training Method

We evaluated the accuracy of incremental training for the two cases described in Section IV-D.

1) Recommendations for Registered Users: The objective of this experiment was to assess the decrease in recommendation accuracy caused by reducing the number of rating scores that were used to construct the base model. In addition, we tried to clarify the differences in recommendation accuracy between the base model and updated models.

Let us first explain the experimental procedure. We used the rating matrix R_t to prepare a base model and a total of ten updated models. The base model was then constructed by using R_t as training data. The updated models was obtained as follows.

1) A temporary rating matrix R'_t was prepared by randomly masking the $K\%$ $(K = 0, 10, 20, \ldots, 90)$ of actual scores in R_t. If K is zero, R'_t is equal to R_t.
2) A temporary base model was built by using R'_t as training data.
3) An updated model was obtained by adding the masked scores, i.e., by using R_t.

Each model was used to rank the musical pieces. To calculate the recommendation accuracies, we used the evaluation matrix R_e for all the settings. We iterated these procedures ten times while switching the rating matrices that were prepared for tenfold cross validation described in Section V-B.

Fig. 8 plots the results. Our method can promptly and appropriately adapt recommendations according to the increase in rating scores. We found that the accuracy barely deteriorated even when the number of rating scores used to update the base model was increased to the number for building it $(K = 50)$. An interesting fact is that the difference in accuracy between

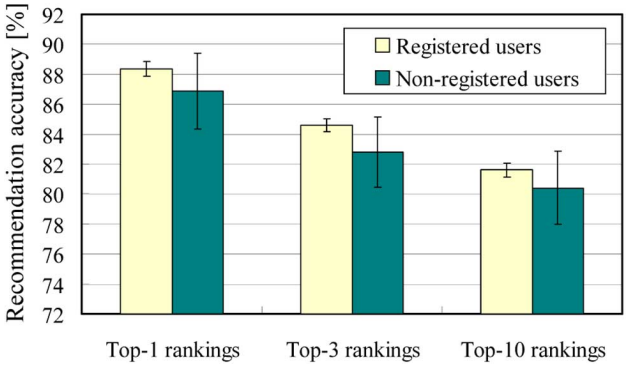

Fig. 9. Recommendation accuracy for registered and nonregistered users.

the base and updated models narrowed as the number of recommended pieces (x) increased. The largest difference was about 5% in evaluating the top-1 rankings $(x = 1)$. However, the recommendation accuracy was sufficiently high even in this case and was higher than the other "nonincremental" methods listed in Table II.

2) Recommendations for Nonregistered Users: The objective of this experiment was to compare the accuracies of recommendations given to registered users with those given to nonregistered users. Smaller differences in recommendation accuracy indicate better performance.

The experimental procedure using the training matrix R_t is as follows.

1) 10% of users U_{new} were randomly selected from U. They were regarded as nonregistered users. We let U_{reg} be the remaining users (registered users).
2) A reduced training matrix R'_t was obtained by removing U_{new} from R_t. Therefore, the size of R'_t was 90% of that of R_t.
3) A temporary base model was constructed by using R'_t as training data.
4) To calculate the recommendation accuracy for U_{reg}, we did the following:
   a) Profiles of U_{reg} in the base model were updated by using R'_t again.
   b) Recommendations based on the updated profiles were evaluated by using the rating scores of U_{reg} in evaluation matrix R_e.

   To calculate the recommendation accuracy for U_{new}, we did the following:
   a) Profiles of U_{new} were created by using the rating scores of U_{new} that were removed in Step 2).
   b) Recommendations based on the created profiles were evaluated by using the rating scores of U_{new} in R_e.

We iterated these procedures ten times while switching the rating matrices that were prepared for the tenfold cross validation described in Section V-B. We repeated the experiment four times and computed the average and variance of the recommendation accuracies.

Fig. 9 clearly shows that our method can make accurate recommendations to nonregistered users as well as to registered users. We found differences in the variance of accuracy in all cases through the F-test at a significant level of 5% ($F(3,3) = 29.4$, $p = 0.02$, $F(3,3) = 26.9$, $p = 0.02$, and $F(3,3) = 29.0$, $p = 0.02$ in the cases of $x = 1, 3, 10$). This is a reasonable observation because rating scores given by nonregistered users were not used to train the base models. Therefore, the recommendation accuracy for nonregistered users tended to be unstable. However, by doing the t-test, we found that there were no differences in average accuracy in any of the cases ($t(3.20) = 0.98$, $p = 0.39$, $t(3.22) = 1.32$, $p = 0.27$, and $t(3.20) = 0.82$, $p = 0.47$ in the cases of $x = 1, 3, 10$).

G. Discussion

We would like to discuss the computational time required to obtain the base and updated models using a standard computer with a 3-GHz Pentium-4 processor. The system required about 10 min to obtain the base model with the EM-based training method. In contrast, the system required only 5 s to update the base model, i.e., to obtain an updated model, with the incremental training method.

There are three remaining issues.

1) The current system cannot incrementally register nonregistered pieces (e.g., new releases) to the three-way aspect model because the current system only takes into account the addition of new scores or nonregistered users. We expect that this problem can be solved by introducing a model-updating formula in the same way as described in Section IV-C.
2) It is necessary to determine until when the model can be incrementally updated, i.e., the point at which the decrease in recommendation accuracy exceeds the user's tolerance. To determine an optimized timing for retraining the whole model, we plan to conduct a user study by deploying our system on realistic large databases.
3) It is important to examine whether semantic properties of latent variables are similar to those of existing genres. To do this, it would be better to use realistic large databases having large variety of genres and moods.

VI. Conclusion

This paper presented a hybrid music recommender system that ranks musical pieces by comprehensively considering collaborative and content-based data, i.e., rating scores derived from users and acoustic features derived from audio signals. To create our system, we used a probabilistic generative model called a three-way aspect model. This model can theoretically explain the generative mechanism for both kinds of the observed data by introducing a set of latent variables, which conceptually correspond to genres. One possible interpretation of the generative mechanism is that a user stochastically selects a genre according to his or her preferences and then the genre stochastically generates a musical piece and an acoustic feature. That is, the joint probability distribution over users, pieces, and features is decomposed into three independent distributions, which are respectively conditioned by genres. These distributions are statistically estimated so that the probability of generating the observed data is maximized. This allows us to incrementally train the aspect model according to the increase in users and rating scores at low computational cost, i.e., we only need to partially update the parameters.

The main contributions can be summarized as follows.

1) We proposed a hybrid recommender system that has three fundamental capabilities: a high degree of recommendation accuracy, a large variety of artists, and a capability for recommending nonrated pieces.
2) We proposed an incremental training method that satisfies an important requirement: a prompt response without deteriorating accuracy.
3) We proposed a bag-of-timbres model that represented time series of MFCCs as a single vector. Our flexible method can be applied to various audio signals or to the time series of various musical features.
4) We demonstrated these capabilities and proved the effectiveness of our bag-of-timbres model by conducting experiments that used real rating scores.

In the future, we plan to use various audio-based features such as tempi, pitches, and rhythmic patterns for improving the representation of the musical content. We may try several toolkits such as MARSYAS [33] and CLAM [34] for automatic feature extraction. In addition, we plan to apply our method to a social networking service (SNS) in which users are introduced to others with similar musical preferences.

ACKNOWLEDGMENT

The authors would like to thank Dr. H. Asoh (AIST, Japan) for his insightful comments.

REFERENCES

[1] A. Uitdenbogerd and R. van Schyndel, "A review of factors affecting music recommender success," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2002, pp. 204–208.
[2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
[3] R. Typke, F. Wiering, and R. Veltkamp, "A survey of music information retrieval systems," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2005, pp. 153–160.
[4] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Computer-Supported Cooperative Work*, 1994, pp. 175–186.
[5] U. Shardanand and P. Maes, "Social information filtering: Algorithms for automating word of mouth," in *Proc. ACM CHI'95 Conf. Human Factors Comput. Syst.*, 1995, pp. 210–217.
[6] J. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proc. 14th Conf. Uncertainty Artif. Intell. (UAI)*, 1998, pp. 43–52.
[7] W. Cohen and W. Fan, "Web-collaborative filtering: Recommending music by crawling the web," *Comput. Netw.*, vol. 33, no. 1–6, pp. 685–698, 2000.
[8] G. Linden and B. Smith, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, Jan./Feb. 2003.
[9] K. Hoashi, K. Matsumoto, and N. Inoue, "Personalization of user profiles for content-based music retrieval based on relevance feedback," *Proc. ACM Multimedia*, pp. 110–119, 2003.
[10] B. Logan, "Music recommendation from song sets," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2004, pp. 425–428.
[11] O. Celma, M. Ramirez, and P. Herrera, in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2005, pp. 464–467.
[12] B. Logan, "Content-based playlist generation: Exploratory experiments," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2002, pp. 295–296.
[13] A. Popescul, L. Ungar, D. Pennock, and S. Lawrence, "Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments," in *Proc. 17th Conf. Uncertainty Artif. Intell. (UAI)*, 2001, pp. 437–444.
[14] T. Hofmann and J. Puzicha, "Latent class models for collaborative filtering," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 1999, pp. 688–693.
[15] T. Hofmann, "Probabilistic latent semantic analysis," in *Proc. 15th Conf. Uncertainty Artif. Intell. (UAI)*, 1999, pp. 289–296.
[16] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *R. Statist. Soc. B*, vol. 39, pp. 1–38, 1977.
[17] R. Mooney and L. Roy, "Content-based book recommending using learning for text categorization," in *Proc. ACM Conf. Digital Libraries*, 2000, pp. 195–204.
[18] D. Billsus and M. Pazzani, "User modeling for adaptive news access," *User Modeling and User-Adapted Interaction*, vol. 10, no. 2–3, pp. 147–180, 2000.
[19] T. Tran and R. Cohen, "Hybrid recommender systems for electronic commerce," in *Proc. Knowledge-Based Electronic Markets, Papers From the AAAI Workshop*, 2000, Tech. Rep. WS-00-04, AAAI Press.
[20] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin, "Combining content-based and collaborative filters in an online newspaper," in *Proc. ACM SIGIR Workshop Recommender Syst.*, 1999.
[21] M. Balabanovic and Y. Shoham, "Fab: Content-based, collaborative recommendation," *Commun. ACM*, vol. 40, no. 3, pp. 66–72, 1997.
[22] M. Pazzani, "A framework for collaborative, content-based, and demographic filtering," *Artif. Intell. Rev.*, vol. 13, no. 5–6, pp. 393–408, 1999.
[23] R. Stenzel and T. Kamps, "Improving content-based similarity measures by training a collaborative model," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2005, pp. 264–271.
[24] P. Melville, R. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering for improved recommendations," in *Proc. Nat. Conf. Artif. Intell. (AAAI)*, 2002, pp. 187–192.
[25] C. Hayes, "Smart radio: Building community-based Internet music radio," Ph.D. dissertation, Trinity College Dublin, Dublin, U.K., 2003.
[26] K. Yu, A. Schwaighofer, V. Tresp, X. Xu, and H.-P. Kriegel, "Probabilistic memory-based collaborative filtering," *IEEE Trans. Knowl. Data Eng.*, vol. 16, no. 1, pp. 56–69, Jan. 2004.
[27] L. Zhang, C. Li, Y. Xu, and B. Shi, "An efficient solution to factor drifting problem in the pLSA model," in *Proc. Int. Conf. Comput. Inf. Technol. (CIT)*, 2005, pp. 175–181.
[28] J.-J. Aucouturier, F. Pachet, and M. Sandler, "'The way it sounds': Timbre models for analysis and retrieval of music signals," *IEEE Trans. Multimedia*, vol. 7, no. 6, pp. 1028–1035, Dec. 2005.
[29] J.-J. Aucouturier and F. Pachet, "Musical genre: A survey," *New Music Res.*, vol. 32, no. 1, pp. 83–93, 2003.
[30] N. Ueda and R. Nakano, "Deterministic annealing EM algorithm," *Neural Netw.*, vol. 11, no. 2, pp. 271–282, 1998.
[31] G. Arfken, "Lagrange multipliers," in *Mathematical Methods for Physicists*, 3rd ed. New York: Academic, 1985, pp. 945–950.
[32] M. Zadel and I. Fujinaga, "Web services for music information retrieval," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2004, pp. 478–483.
[33] G. Tzanetakis and P. Cook, "MARSYAS: A framework for audio analysis," in *Organized Sound*. Cambridge, U.K.: Cambridge Univ. Press, 2000.
[34] X. Amatriain, J. Massaguer, D. Garcia, and I. Mosquera, "The CLAM annotator: A cross-platform audio descriptors editing tool," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, 2005, pp. 426–427.

Kazuyoshi Yoshii (S'05) received the B.E. degree and the M.S. degree in Informatics from Kyoto University, Kyoto, Japan, in 2003 and 2005, respectively. He is currently pursuing the Ph.D. degree in the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University. He is supported by the JSPS Research Fellowships for Young Scientists (DC1).

His research interests include music scene analysis, music recommendation, and human–machine interaction.

Mr. Yoshii received nine awards including the IPSJ Yamashita SIG Research Award, the FIT 2004 Paper Award, the Interaction 2006 Best Presentation Award, and the Best-in-Class Award of MIREX 2005. He is a member of the Information Processing Society of Japan (IPSJ) and the Institute of Electronics, Information, and Communication Engineers (IEICE).

Masataka Goto received the Doctor of Engineering degree from Waseda University, Tokyo, Japan, in 1998.

He then joined the Electrotechnical Laboratory (ETL), which was reorganized as the National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan, in 2001, where he has been a Senior Research Scientist since 2005. He served concurrently as a Researcher in Precursory Research for Embryonic Science and Technology (PRESTO), Japan Science and Technology Corporation (JST), from 2000 to 2003, and has been an Associate Professor in the Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba, since 2005.

Dr. Goto received 21 awards, including the IPSJ Best Paper Award, IPSJ Yamashita SIG Research Awards, and Interaction 2003 Best Paper Award. He is a member of the IPSJ, ASJ, JSMPC, IEICE, and ISCA.

Kazunori Komatani received the B.E. degree, M.S. degree in Informatics, and the Ph.D. degree, all from Kyoto University, Kyoto, Japan, in 1998, 2000, and 2002, respectively.

He is currently an Assistant Professor in the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University. His research interests center on spoken language processing, especially on spoken dialogue systems.

Dr. Komatani has received several awards including the 2002 FIT Young Researcher Award and 2004 IPSJ Yamashita SIG Research Award from the Information Processing Society of Japan (IPSJ), and the RSJ/SICE Award for IROS-2006 Best Paper Nomination Finalist. He is a member of the IPSJ, Institute of Electronics, Information and Communication Engineers (IEICE), Association for Natural Language Processing (NLP), Japanese Society for Artificial Intelligence (JSAI), Association for Computational Linguistics (ACL), and International Speech Communication Association (ISCA).

Tetsuya Ogata (M'00) received the B.S., M.S., and D.E. degrees in mechanical engineering from Waseda University, Tokyo, Japan, in 1993, 1995, and 2000, respectively.

From 1999 to 2001, he was a Research Associate with Waseda University. From 2001 to 2003, he was a Research Scientist in the Brain Science Institute, RIKEN. Since 2003, he has been a Faculty Member in the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University, Kyoto, Japan, where he is currently an Associate Professor. Since 2005, he has been a Visiting Associate Professor in the Humanoid Robotics Institute, Waseda University. His research interests include human–robot vocal-sound interaction, dynamics of human–robot mutual adaptation, and active sensing with robot systems.

Dr. Ogata received the 2000 JSME Outstanding Paper Medal from the Japan Society of Mechanical Engineers, the Best Paper Award of IEA/AIE-2005, and the RSJ/SICE Award for IROS-2006 Best Paper Nomination Finalist. He is a member of the IPSJ, JSAI, RSJ, JSME, HIS, and SICE.

Hiroshi G. Okuno (SM'06) received the B.A. and Ph.D. degrees from the University of Tokyo, Tokyo, Japan, in 1972 and 1996, respectively.

He worked for Nippon Telegraph and Telephone, Kitano Symbiotic Systems Project, and Tokyo University of Science. He is currently a Professor in the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University. He was a Visiting Scholar at Stanford University, Stanford, CA, and a Visiting Associate Professor at the University of Tokyo. He has done research in programming languages, parallel processing, and reasoning mechanisms in AI, and he is currently engaged in computational auditory scene analysis, music scene analysis, and robot audition. He edited with D. Rosenthal *Computational Auditory Scene Analysis* (Lawrence Erlbaum Associates, 1998), with T. Yuasa *Advanced Lisp Technology* (Taylor and Francis, 2002), and with M. Ali *New Trends in Applied Artifical Intelligence* (Springer-Verlag, 2007).

Dr. Okuno has received various awards including the 1990 Best Paper Award of JSAI, the Best Paper Awards of IEA/AIE-2001 and 2005, IEEE/RSJ Nakamura Award for IROS-2001 Best Paper Nomination Finalist, and the RSJ/SICE Award for IROS-2006 Best Paper Nomination Finalist. He was also awarded the 2003 Funai Information Science Achievement Award. He is a member of the IPSJ, JSAI, JSSST, JSCS, RSJ, ACM, AAAI, ASA, and ISCA.